# SESSION

# USE OF ROBOTS, GAMIFICATION, AND OTHER TECHNOLOGIES IN EDUCATION

## Chair(s)

**TBA**

# Exploring Finite State Automata with Junun Robots: A Case Study in Computability Theory

Vladimir Kulyukin Sarat Kiran Andhavarapu Melodi Oliver Christopher Heaps
Soren Rasmussen Wesley Rawlins
Department of Computer Science
Utah State University
Logan, UT, USA
vladimir.kulyukin@usu.edu, saratkiran18@gmail.com, melodi_oliver@hotmail.com, heaps.chris@gmail.com, soren.rasmussen@aggiemail.usu.edu, wesleyrawlins@icloud.com

***Abstract*— A case study in computability theory is presented on teaching finite state automata with mobile robots. Junun Mark III robots were used in several coding assignments on finite state machines and a final coding project in an upper-level undergraduate course in computability theory. Software and hardware choices are discussed with respect to the robotic platform and the laboratory environment. Several guidelines are presented for integrating robots in theory of computation classes to reinforce and enhance students' understanding of finite state machines through practical applications of finite state control mechanisms.**

***Keywords—computability; undergraduate university education; mobile robots; theory of computation; finite state automata***

## I. Introduction

Computability theory, also known as theory of computation (TOC), remains one of the most challenging Computer Science (CS) courses both for professors and students. Professors, who teach computability courses, face lack of motivation on the part of the students as well as inadequate mathematical maturity and problem solving skills [1]. Students, who enroll in TOC courses, especially when such courses are required, believe that the covered topics are marginally relevant to having successful IT careers or, even when relevant, the topics are not well motivated in the course materials. Although many students later realize the fundamental and transcendental nature of concepts covered in TOC courses, they frequently fail to realize it as they are actually taking the course due to abundant technical notation, complicated terminology, and the tendency of some professors to overformalize intuitively straightforward concepts.

Another problem faced by CS undergraduates is that they typically take TOC courses in their senior year when they are almost finished with the undergraduate curriculum. The first author of this paper has been teaching an undergraduate TOC class at Utah State University (USU) for the past seven years and has heard many CS majors telling him how much easier the advanced courses on compiler construction, game development, and programming languages would have been, had they taken a TOC class earlier in their undergraduate curriculum.

The use of robots in undergraduate CS education has gained a significant interest both among researchers and educators [2, 3, 4]. This interest could be attributed to the fact that robotics is an inherently multi-disciplinary area bringing together subjects as diverse as cognitive psychology, electrical engineering, and computer vision. It has been reported in the literature that the use of robots improves student project engagement [5].

We also believe that the use of robots can improve student engagement in and appreciation of not only the areas traditionally affiliated with robotics (e.g., computer vision and electrical engineering) but also theoretical computer science. Specifically, mobile robots present an effective opportunity to motivate topics and concepts seemingly detached from any practical applications. Robots can also be used in conjunction with educational software tools such as JFLAP (www.jflap.org) to illustrate and motivate the practical significance of theoretical concepts.

The effective use of robots in the classroom may require special training and hardware. Some professors who teach TOC courses and students who take TOC electives tend to be comfortable only with the topics for which pencil and paper are the most adequate technology. Some professors and students may feel intimidated by the apparent complexity and diversity of robotic platforms.

Fortunately, the advances in commercial robotic platforms have made the integration of robots into TOC courses much easier than it used to be a decade ago (e.g., www.activrobots.com, www.robotshop.com, www.junun.org). There are two key decisions faced by CS professors who want to use robots in TOC course: the choice of a robotic platform and the choice of topics that the robots will be used to illustrate, explore, or motivate. While there are many outstanding books on computability [6, 7, 8] and various theoretical aspects of robotics [9, 10], there is a relative shortage of practical step-by-step resources that CS instructors can use to integrate robots into TOC courses.

In this paper, we contribute to overcoming this shortage of step-by-step materials by sharing some of our experiences of using Junun robots to motivate, illustrate, and explore the topic of finite state automata (FSA). These experiences are primarily based on the TOC course taught by the first author in the fall 2014 semester. The second author was the teaching assistant for this course. The other four authors are USU CS undergraduate students who were enrolled in the course.

The remainder of our paper is organized as follows. In Section II, we present the basic theoretical concepts for which we used mobile robots. These concepts were covered in the fall 2014 TOC course (CS 5000) taught by the first author at USU. CS 5000 used to be a required course with an average student enrollment of approximately forty students per semester until 2008. The course has been an elective since 2008 with the student enrollment ranging from five to fifteen students. In Section III, we describe the Junun Mark III platform, its modules, and assembly requirements. The three text books were used for the course: the book by Davis et al. [6] was used for computability; Brooks Webber's text [7] and the book by Hopcroft and Ullman [8] were used for finite state machines. In Section IV, we present several coding assignments given to the students and the final project of the course. Section V summarizes our experiences, analyzes our successes and failures, and presents recommendations that interested CS instructors or students may want to consider if they want to integrate mobile robots into their coursework.

## II. Basic Concepts

In this section, we present several formal concepts enhanced and illustrated with mobile robots in our computability course. These concepts are related to the basic notion of the finite state automaton (FSA). We chose the FSA, because there is a natural fit between robotic control and finite state machines that has been successfully and repeatedly illustrated in many research projects on mobile robots [9, 10].

### A. Finite State Automata as Directred Graphs

In an introductory lecture, a finite state automaton (FSA) was informally described as a directed graph whose nodes are states and whose edges are transitions on specific symbols. We pointed out to the students that input symbols processed by the FSA can be as simple as 0 and 1 or as complex as robot action sequences. For example, suppose we want to design a control system for a robotic camera and arm unit for building two-block towers from blocks *A* and *B*, as shown in Figure 1.

We can arbitrarily assume that the sensor-motor system of the unit can execute one action, $Puton(X,Y)$, and two predicates: $Clear(X,Y)$ and $On(X,Y)$. The action $Puton(X,Y)$ places block *X* on block *Y* or on table *T*, $Clear(X,Y)$ verifies that the top of block *X* is clear, and $On(X,Y)$ verifies that block *X* is either on top of another block or on top of the table *T*.

Figure 2 shows a finite control for the two-block domain. The control consists of three states Q0, Q1, and Q2. Each state is a set of predicates that describe a state of the world. In state Q0, both blocks are on the table. In state Q1, *A* is on *B*. In state Q2, *B* is on *A*. The camera-arm unit can start in any of the three states. For example, the unit can start in Q1 shown in Figure 1 and reach the state of the world in Q2 by executing $Puton(A,T)$ and $Puton(B,A)$.

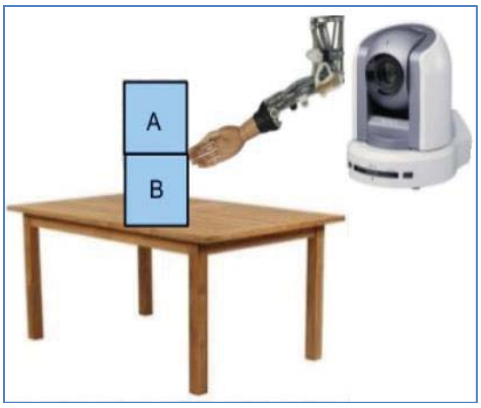



**Figure 1. A robotic camera-arm unit**

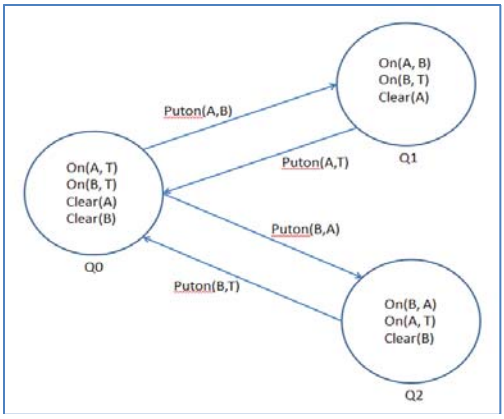



**Figure 2. Finite state control for two-block domain**

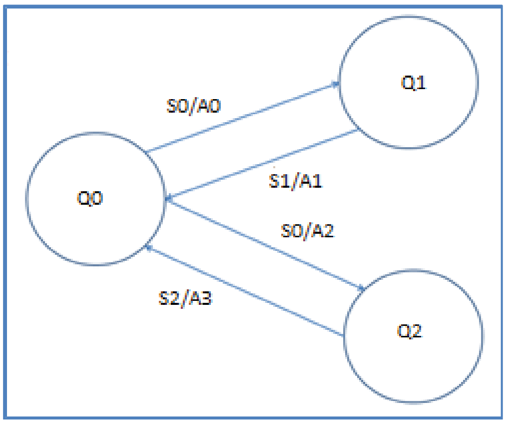



**Figure 3. A Mealy Machine**

A well-known limitation of the FSA is that the output is limited to the binary decision of acceptance or rejection. The FSA can be modified by choosing the output from an alphabet other than the input, which brings us to the concepts of the Mealy and Moore machines [8]. The Moore machine associates outputs with states whereas the Mealy machine associates outputs with transitions. In both types of automata, the outputs can be interpreted as commands to robot hardware. Figure 3 gives an example of how a finite state machine shown in Figure 2 can be transformed into a Mealy machine.

In Figure 3, The $X/Y$ notation means that when the machine sees $X$ in the input, it outputs $Y$ and transitions to a new state. In Figure 3, we have defined the following complex symbols: *S0* = {*On(A,T)*, *On(B,T)*, *Clear(A)*, *Clear(B)*}, *S1* = {*On(A,B)*, *On(B,T)*, *Clear(A)*}, *S2* = {*On(B,A)*, *On(A,T)*, *Clear(B)*}, *A0* = *Puton(A,B)*, *A1=Puton(A,T)*, *A2=Puton(B,A)*, *A3=Puton(B,T)*. Thus, when the robot's sensory system recognizes the complex symbol *S0*, the robot sends *A0* command to its hardware for execution.

*B. Formal Definitions*

After the students were informally exposed to the regular FSA and their modifications such as the Moore and Mealy machines, two lectures were dedicated to formalizing these concepts and proving several fundamental theorems. In this section, we briefly review these formalizations without going into low-level technical details. Informal feedback from the students indicated that the prior informal introduction of these concepts was helpful and prepared them for more formal lectures and assignments on finite state machines.

Formally, a DFA $M$ is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where $Q$ is a finite set of states, Σ is an input alphabet, δ is a transition function $\delta: Qx\Sigma \rightarrow Q$, $q_0$ is the start state and $F$ is the set of final states. To process sequences of symbols, we can define another transition function $\delta^*(q, \varepsilon) = q$; $\delta^*(q, xa) = \delta(\delta^*(q, x), a)$, where $x \in \Sigma^*$ and $a \in \Sigma$.

For example, for the finite control unit shown in Figure 2, Σ={$Puton(A, B), Puton(A, T), Puton(B, A), Puton(B, T)$} and δ can be defined through equalities such as $\delta\big(Q0, Puton(A, B)\big) = Q1$. It should be noted that the finite control in Figure 2 is not, technically speaking, a DFA in that it does not specify the unique start state or define all possible transitions for every state. For example, the transition *δ(Q1,Puton(B, A))* is not defined.

A string $x$ is accepted by $M$ if and only if $\delta(q_0, x) \in F$. The language accepted by $M$ is $L(M) = \{x|\delta^*(q_0, x) \in F\}$. If we define *Q0* to be the start state and make all three states final, then the sequence $Puton(A, B), Puton(A, T), Puton(B, A)$ is in the language of the control unit in Figure 2.

We can define finite state automata for which there can be more than one sequence of legal states to accept the same input. Such automata are called non-deterministic finite automata (NFA). An NFA accepts an input so long as there is one legal sequence of states that begins at the start state and ends in a final state. Formally, an NFA $M$ is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$, where all symbols are defined exactly as they are in the case of DFA except for the transition function which maps pairs of states and symbols into the power set of $Q$: $\delta: Qx(\Sigma\ U\ \{\varepsilon\}) \rightarrow P(Q)$, where $P(Q)$ is the power set of $Q$.

After the students were introduced to a few examples of NFAs, we proved that an NFA with ε-transitions can be transformed into an equivalent NFA without ε-transitions. Then the students were given a few examples showing how a DFA could keep track of the states that a given NFA can be in after reading each symbol of the input. The students were successfully led to the discovery that since the NFA can be in more than one state after reading a symbol, each state of an equivalent DFA must correspond to a subset of the NFA's states. After the students made that discovery, we proved the theorem about the equivalence of DFA and NFA by using the subset construction [7].

We concluded the formalization of the finite state automata with a lecture on the Mealy and Moore machines. A Moore machine $M$ is a six-tuple $(Q, \Sigma, \Delta, \delta, \lambda, q_0)$, where $Q, \Sigma, \delta, q_0$ are as in the DFA while $\Delta$ is the output alphabet and $\lambda$ is a mapping from $Q$ to $\Delta$. This $\lambda$-mapping defines the output for each state. If a Moore machine $M$ goes through a sequence of states $q_0, q_1, \dots, q_n$, the output of $M$ is $\lambda(q_0)$,$\lambda(q_1), \dots, \lambda(q_n)$.

A Mealy machine M is also a six-tuple $(Q, \Sigma, \Delta, \delta, \lambda, q_0)$ where all symbols are defined as in the Moore machine except that $\lambda$ is defined as a mapping $Qx\ \Sigma \rightarrow \Delta$. In other words, $\lambda$ associates a transition from a specific state on a specific symbol with an output symbol. The output of the Mealy machine $M$ on an input $a_1, a_2, \dots, a_n$ is $\lambda(q_0, a_1), \lambda(q_1, a_2), \dots, \lambda(q_{n-1}, a_n)$. For example, for the Mealy machine in Figure 3, $\Sigma = \{S0, S1, S2\}$ and $\Delta = \{A0, A1, A2\}$.

Both Moore and Mealy machines can be used to implement various robotic control structures to map sensory data to hardware actions. Whether outputs are associated with states or transitions is mostly implementation-dependent. The lecture on the Moore and Mealy machine was concluded with an informal sketch of the proof of the theorem that the Moore and Mealy machines are equivalent [8].

## III. Junun Mobile Robots

*A. Hardware*

After the theoretical introduction into the finite state machines was concluded, one lecture was dedicated to an introductory robotics laboratory. None of the students had worked with robots before and we wanted to give them a brief introduction to robotics and some hands-on experience.

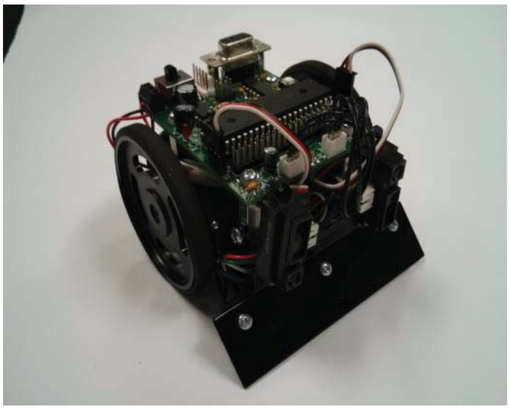


**Figure 4. Assembled Junun Mark III robot**

The students came to our laboratory where the hardware basics of the Junun Mark III robotic platform (see Figure 4) were explained to them. They were also shown how to connect to the robot from a Windows computer, compile simple robot programs, upload them to the robot and run the robot. The students had an opportunity to play with robots and to write their own JAL programs. JAL, an acronym for **J**ust **A**nother

Language, is a Pascal-like programming language created by Wouter van Ooijen for several PIC microcontrollers.

As we have previously stated, there are many robotic platforms available to CS instructors. We chose the Junun Mark III robot platform (http://www.junun.org) for its modularity, hardware and software adaptability, and a low budget. We purchased a total of five unassembled Mark III kits and assembled them in the laboratory. Each kit was approximately ninety dollars.

The main components of the Mark III robot hardware are the controller and the chassis, shown in Figure 5, respectively. The controller board, shown in Figure 5 (left), holds a microcontroller, a power regulator, a serial interface, and connection points for the off-board sensors. Most of the work to build Mark III robots went into assembling and soldering the controller board.

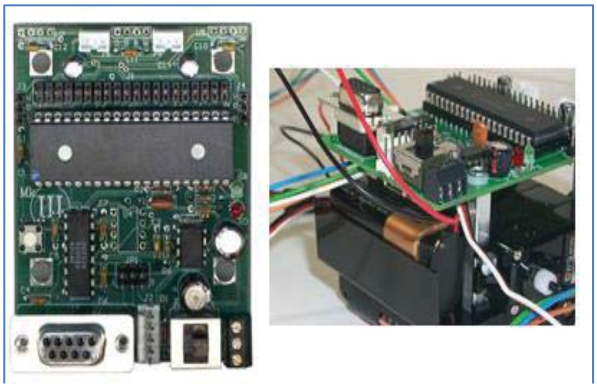


**Figure 5. Controller board (left); Assembled chassis (right)**

The chassis, shown in Figure 5 (right), is the mechanical base for the Mark III robot where the motors are mounted. The chassis holds the electronics and sensors in place and consists of a base plate, a scoop, four standoffs, a 4AA cell battery holder, a 9V battery snap, and assorted fasteners to hold everything together.

Each robotic kit took nearly a day to assemble. The most challenging assembly aspects were modifying the servo motors for continuous rotation, soldering the densely populated PCB, and calibrating the sensors.

CS professors wishing to integrate Mark III robots into their curriculum should keep in mind that soldering is a must. While the platform is relatively robust and, when assembled, performs according to the specifications, it is not as plug-and-play as we hoped it would be. The tools necessary for Junun assembly are a soldering iron, solder, wire cutters, needle nose pliers, utility knife/X-Acto, a jeweler's screwdriver, a digital multi-meter, and a drill.

### B. Software

Junun Mark III robots can be programmed in 16f84a Assembler, which is possible but not that easy. JAL is a higher level alternative available for the platform. The JAL compiler is freely available from jal.sourceforge.net under GPL. The compiler converts the Jal source code into hexadecimal files that can run directly on the robot. JAL is a higher-level language for a number of PIC microcontrollers such as 16c84 and 16f84. The JAL compiler is available on Linux, Mac OS X, and Windows.

JAL is a straightforward language to learn for anyone who has had some programming experience with imperative programming languages. The only variable type is byte. The language has loops and return statements. The students were given several examples of JAL programs, e.g., an implementation of the Fibonacci numbers, and had a chance to compile them with the JAL compiler installed on a Windows machine in the laboratory.

## IV. Coding Assignments

We gave several coding assignments to the students to implement various finite state control mechanisms on Junun robots. At the end of the semester we held a line following and obstacle avoidance competition. In this section, we will describe each coding assignment, the final competition, and analyze our successes and failures. The coding assignments presented in this section were not the only assignments in the course. There were a total of ten assignments, approximately five of which were theoretical and the other five programmatic.

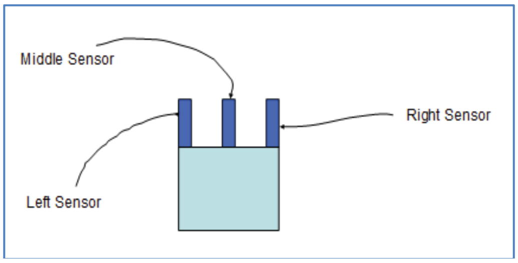



**Figure 7. Three line detection sensors**

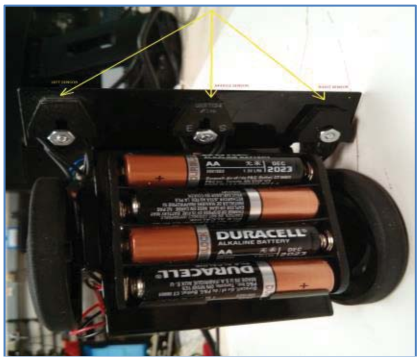


**Figure 8. Real line detection sensors**

### A. Line Following

Line following is a standard benchmark for many robot control mechanism. For the line following assignment, the Junun robot was abstractly described to the students as a module of three downward-looking infrared sensors, as shown in Figure 7. We showed the students the actual hardware equivalents of these sensors marked in Figure 8 with yellow arrows. Each sensor is a Fairchild QRB1134 IR two-element photo-reflector. One element emits a beam of infrared light and the other detects that light. The light is reflected by an object in front of the sensor. The sensor is robust in that it has a built-in daylight filter to reduce interference from ambient lighting. The optimal sensing distance, according to the documentation, is about 0.2 inches, which makes it suitable for detecting dark continuous lines on the floor.

Figure 9 shows three possible situations that a robot can face while following the line. In the left situation, the left sensor detects the line while the middle and the right sensors do not. The robot must take a corrective left turn to ensure that its middle sensor senses the line. In the middle situation, the robot's middle sensor senses the line so the robot can continue to move forward. In the right situation, only the robot's right sensor senses the line whereas the left and middle sensors do not. The robot must take a corrective right turn to put its middle sensor back on the line.

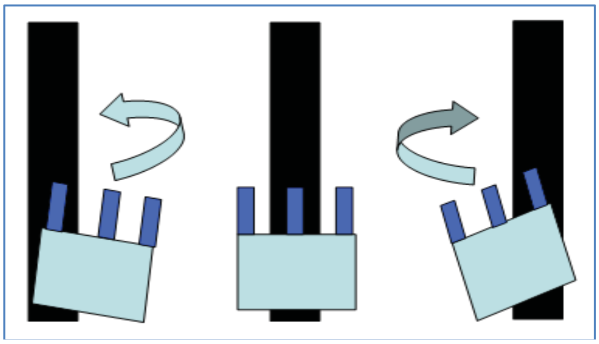


**Figure 9. Three line following situations**

Figure 10 gives a JAL implementation of the three line following situations outlined in Figure 9. The variables adc8bitL, adc8bitC, and adc8bitR hold the readings for the left, middle (center), and right photo-reflectors, respectively. A value of 128 is an experimentally discovered threshold at which the darkness is reliably detected on our Junun robots. Other Junun robots may work with a different threshold. The arguments to the functions **turn_left**, **move_forward**, and **turn_right** are hardware delays measured in milliseconds. The longer the delay the more the robot turns or moves. The last **turn_left** command is a corrective action because we discovered that our assembled robots tended to veer slightly right.

```
procedure take_line_follow_action is
  if ( adc8bitL > 128 )
    then turn_left(25)
  elsif ( adc8bitC > 128 )
    then move_forward(1)
  elsif ( adc8bitR > 128 )
    then turn_right(25)
  else
     turn_left(5)
   end if
end procedure
```

**Figure 10. Three line following situations**

A finite state control for the line following problem is given in Figure 11 where the nodes denote the hardware commands and the transitions are the sensor outputs. Another possibility is to have states correspond to the sensor outputs and hardware commands to transitions, as is the case with Mealy machines.

Figure 12 gives a JAL implementation of the finite state control for the line following control. The procedure reads the three sensors and calls the **take_line_follow_action** function coded in Figure 10.

*B. Finding and Following a Line*

The next programming assignment was to build up on the line following assignment to implement a finite state control that enables the robot placed inside a polygon to find a line and follow it. Figure 13 shows a possible solution to this problem. Since the robot is placed inside a polygon defined by a continuous line, the robot can simply choose a direction in which it is facing or choose a random direction and then start moving slowly in that direction. This movement continues so long as no line is detected. When the line is detected, the control goes to the automaton that follows the detected line.

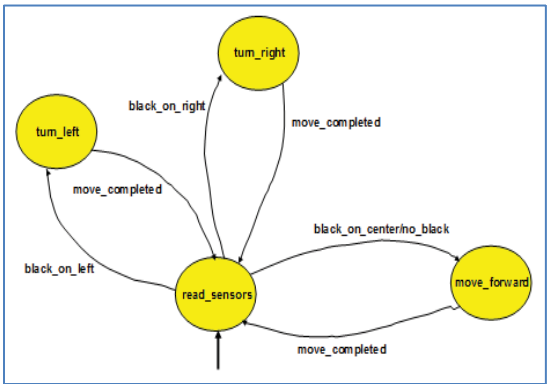



**Figure 11. A Possible line following control**

```
procedure follow_line is
   forever loop
      -- read left line sensor
      f877_adc_8(0, 7, adc8bitL)
      -- read center line sensor
      f877_adc_8(0 ,6 , adc8bitC)
      -- read right line sensor
      f877_adc_8(0 ,5 , adc8bitR)
      take_line_follow_action
   end loop
end procedure
```

**Figure 12. A JAL implementation of line following**

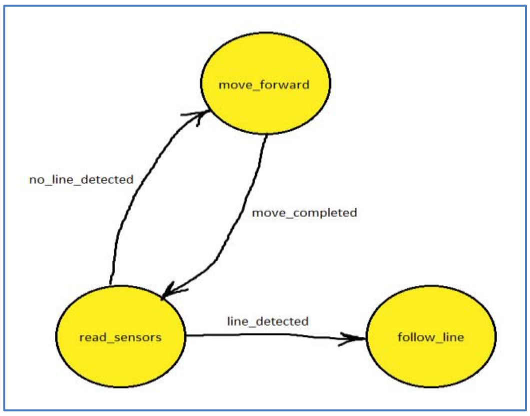



**Figure 13. Find-and-follow-line control**

```
procedure seek_line is
  if ( adc8bitL < 128 ) & ( adc8bitC < 128 ) &
     (adc8bitR < 128 )
         then  move_forward(2)
  else
          take_line_follow_action
  end if
end procedure

procedure find_and_follow_line is
   forever loop
         f877_adc_8(0, 7, adc8bitL)
         f877_adc_8(0 ,6 , adc8bitC)
         f877_adc_8(0 ,5 , adc8bitR)
         seek_line
   end loop
end procedure
```

**Figure 14. JAL implementation of find-and-follow-line**

Figure 14 shows a JAL implementation of the find-and-follow-line control in Figure 13. The procedure **seek_line** checks the readings of the three downward-looking photo-reflectors. If all three sensors do not detect darkness, the robot keeps moving forward. Once the line is detected, the procedure **take_line_follow_action** is called that keeps the robot on the line. So long as the robot stays on the line, the if condition in **seek_line** is false and **take_line_follow_action** keeps getting called.

*C. Object Following*

The third programming assignment gave the students some experience with the two front end GP2Y0A21YK distance measuring sensors. Each sensor determines the range to the target between 10 and 80 centimeters. Our experiments showed that the most reliable distance is between 15 and 20 cm.

```
forever loop
  f877_adc_8(0 ,2 , eyeR)
  f877_adc_8(0 ,3 , eyeL)
  f877_adc_8(0 ,5 , lineR)
  f877_adc_8(0 ,6 , lineC)
  f877_adc_8(0, 7, lineL)
  var byte leftZeroThresh = 0x09
  var byte rightZeroThresh = 0x09
  var byte seeThresh = 0x55
  if ((eyeL< leftZeroThresh) & (eyeR < rightZeroThresh))
    then robot_stop
  elsif ((eyeL > seeThresh) & (eyeR > seeThresh)) then
      move_forward(1)
  elsif((eyeL > seeThresh) & (eyeR < seeThresh)) then
      turn_left(25)
  elsif((eyeL < seeThresh ) & (eyeR > seeThresh)) then
      turn_right(25)
  end if
end loop
```

**Figure 15. JAL implementation of follow-object control**

The assignment was to design and implement a finite state control for the robot to follow an object by using the two front distance sensors. If the readings of the left and right sensors are below the appropriate thresholds, the robot stops. If the reading of the left sensor is above the threshold and the reading of the right sensor is below the threshold, the robot turns left. If the reading of the left sensor is below the threshold and the reading of the right sensor is above the threshold, the robot turns right. Figure 15 gives a JAL implementation of this finite state control.

*D. Line Following and Obstacle Avoidance*

The final project of the course was to implement a finite state control for the Junun robot to follow a line and avoid small obstacles placed on the line. Figure 16 shows the final project course with two obstacles placed on it.

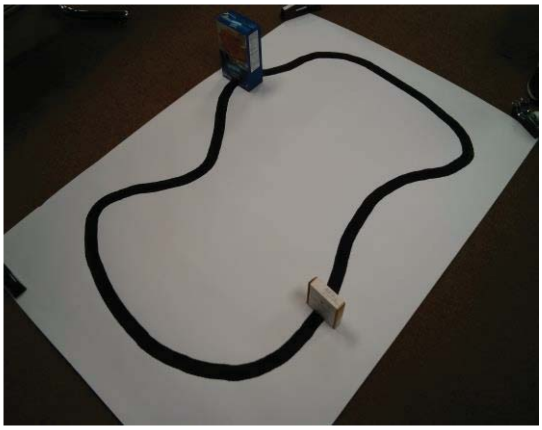


**Figure 16. Line course with obstacles**

When placed on an arbitrary position on the line, the robot starts following the line and follows it until it detects an obstacle. When an obstacle is detected, the robot must do an obstacle avoidance maneuver. A JAL implementation of a simple obstacle avoidance maneuver is shown in Figure 17.

```
procedure avoid_obstacle is
      turn_left(100)
      turn_left(100)
      turn_left(100)
      turn_left(100)
      turn_left(100)
      turn_left(100)
      move_forward(100)
      turn_right(100)
      turn_right(100)
      turn_right(100)
      turn_right(100)
      turn_right(100)
      turn_right(100)
end procedure
```

**Figure 17. JAL implementation of obstacle avoidance**

## V. Discussion

The first two authors conducted regular informal surveys of the students' perception of the material and their attitude to the integration of robots into the curriculum. The first author did it as instructor while the second author did it as teaching assistant.

The overall response of the students was overwhelmingly positive. All students told us that implementing their solutions and trying them on real robots helped them understand the formal definitions and theorems discussed or proved in class.

All students successfully completed the first coding assignment of implementing the line following control. One difficulty that the students pointed out to us was that since there is no JAL-compatible Junun emulator, there was no way for the students to evaluate their code without physically being in the laboratory. The students told us that if they had had an opportunity to run their Jal coding solutions on an emulator they would have completed their coding assignments much faster.

The second coding assignment of finding and following a line was successfully completed by approximately fifty per cent of the students. The students who did not complete this assignment implemented unnecessarily complex line seeking procedures that would turn the robot by small degrees and look for a line. This behavior would cause the robots to get stuck inside the line seeking loop without ever finding the actual line. The students who successfully completed this assignment realized that since the robot was by assumption placed inside a closed polygon the robot should steadily move straight in an arbitrary direction looking for a line.

All students successfully completed the third coding assignment of object following. This assignment was conceptually simpler than the first two coding assignments. The students implemented controls similar to the one given in Figure 15. The objective of this assignment was to give the students enough conceptual and programmatic machinery for the final project.

All students put a great deal of effort into the final project. However, only fifty percent of the students completed the project. All students successfully implemented the line following control and obstacle detection. However, the students who failed to complete the project had difficulty implementing a robust obstacle avoidance control. Some students ended up implementing bull dozing controls that caused the robots to bulldoze obstacles instead of avoiding them.

Finally, we would like to discuss several lessons we have learned as instructors in the hope that both instructors and students will benefit from them. The most important lesson is Junun hardware incompatibility. All five robots had the same components and were assembled in our laboratory in the exact same way. However, we quickly discovered that the same software ran differently on different robots. This caused a grading problem in the first coding assignment when one student was downgraded because his code, although logically correct, caused the robot to quickly veer off the line. The student, however, emailed us a movie that he recorded with his cellphone when the robot successfully followed the line for five minutes. We invited the student to the laboratory and discovered that he had used a different robot to test his code than the robot on which his submitted code was tested and graded. When his coded was uploaded to the robot that he had used to complete the assignment, the robot followed the line without any problems. After that incident, we asked the students to tell us on which robots they implemented and tested their software.

Another lesson is the importance of purchasing rechargeable batteries. All batteries we purchased originally were not rechargeable, which caused logistical problems when the students would come to the laboratory to test their code only to discover that the batteries on all robots were drained by the students who had come before them.

The third lesson is a positive impact that the integration of mobile robots had on the class. Although there were hardware problems and failures, the robots turned out to be an excellent exploration and demonstration tool for the concepts of finite state automata.

## *References*

[1] Surma, D. R. "Rolling the dice on theory: one department's decision to strengthen the theoretical computing aspect of their curriculum." *Journal of Computing Sciences in Colleges*, vol. 28 (1), pp. 74-80, 2012.

[2] Cielniak, G., Bellotto, N., and Duckett, T. "Integrating mobile robotics and vision with undergraduate computer science." *IEEE Transactions on Education*, vol. 56 (1), pp. 48 – 53, 2013.

[3] Maxwell, B.A. and Meeden, L.A. "Integrating robotics research with undergraduate education," *IEEE Intell. Syst. Mag.*, vol. 15, no. 6, pp.22–27, Nov.–Dec. 2000.

[4] Kulyukin, V., Gharpure, C., and Coster, D. "Robot-assisted shopping for the visually impaired: proof-of-concept design and feasiblity evaluation." *Assistive Technology*, vol. 20.2, pp. 86-98, Summer 2008.

[5] Challinger, J. "Efficient use of robots in the undergraduate curriculum." *ACM SIGCSE Bulletin*, 37(1), 436-440, 2005.

[6] Davis, M., Sigal, R., and Weyuker, E.J. *Computability, Complexity, and Languages: Fundamentals of Theoretical Computer Science*, 2nd Edition. Academic Press, New York, 1994.

[7] Brooks Webber, A. Formal Language: *A Practical Introduction*. Franklin, Beedle & Associates, Inc., Wilsonville, Oregon, 2008.

[8] Hopcroft, J.E. and Ullman, J.D. *Introduction to Automata Theory, Languages, and Computation*. Narosa Publishing House, New Delhi, India, 2002.

[9] Arkin, R.C. *Behavior-Based Robotics*. The MIT Press, Cambridge, Massachusets, 1998.

[10] Murphy, R.R. *Introduction to AI Robotics*. The MIT Press, Cambridge, Massachusetts, 2000.

# Meaningful Gamification of a Computer Science Department: Considerations and Challenges

**Marlon Mejias[1], Ketly Jean-Pierre[1], Qi'Anne Knox[1], Elizabeth Ricks[2], Ph.D., Legand Burge[1], III, Ph.D., and A. Nicki Washington[1], Ph.D.**
Department of Computer Science[1], Department of Human Development and Psychoeducational Studies[2], Howard University, Washington, DC, USA

***Abstract –*** *Gamification presents an opportunity to model and guide the enculturation process of computer science undergraduates. It can also make the matriculation journey more engaging for a broader range of students if done correctly. This paper looks at design considerations, potential benefits and the challenges with implementing meaningful gamification in a department of computer science. It describes the underlying theories of motivation and fun in gamification as well as the application of Werbach's gamification framework to gamifying a Computer Science department.*

**Keywords:** Gamification, Computer Science Culture**.**

## 1 Introduction

Research has shown that computer science (CS) students are not prepared for life after graduation. Students lack effective communication and collaborative skills, as well as the technical skills to support large-scale development[1][2]. Research also shows that students have not developed the high level cognitive skills of design [3]. These shortcomings can be attributed to the fact that students are not participating in enough holistic development outside of the classroom. Gamification may be a means of influencing behavioral change that can address these issues.

Gamification as a research topic has increasingly been trending in academia [4] because of its potential to engage students and produce behavioral change [5]. Most of the published research in higher education lacks a theoretical underpinning that can help readers understand the researchers' motivation and the justifications for how their gamification approach is supported by any theory of change. This means that findings are difficult to generalize and don't contribute to the larger body of gamification knowledge.

Gamification is widely defined as "…the use of game design elements in none-game contexts" [6]. Game design elements fall into three categories, dynamics, mechanics and components [7]. Most implementations of gamification research focus on the implementation of game components, with little attention to dynamics and mechanics.

Game dynamics are the targeted behavior, and the emotions that game designers seek to elicit from gameplay [7]. Elements of the game dynamic that should be designed for are: motivators, tradeoffs, progression loops, narrative and the interactions between players. Mechanics are the structure that drive player engagement loops [7]. They consists of the objectives, procedures, and rules. Mechanics include but are not limited to challenges, chance, competition, feedback, resource acquisition, rewards, transactions, and turns and win states. Game components are the specific instantiations of the desired dynamics and mechanics [7]. Available components for gamification are, achievements, avatars, badges, major projects, collections, competitions, content unlocking, gifting, leaderboards, points, levels, quests, social graphs, teams and virtual goods.

Meaningful gamification has been successfully incorporated outside of higher education. Examples include Nike+ and Stack Overflow. These efforts have led to high levels of engagement and in the case of Stack Overflow, engagement in tasks that are highly cognitive. Stack Overflow gives points and badges for answering software related questions by allowing users to vote for the best answers to posted questions. When a user's answer is selected they get a sense of competence and being a useful member of the community when their answer is selected. This approach also employs a variable reward schedule. The user is not guaranteed recognition for each of their submission. This creates a sense of anticipation that heightens the users sense of success [8]. One of the major criticisms that can be applied to most gamification efforts thus far is that it has just been a 'pointsification' [9] or 'exploitationware' [10] approach of focusing on game components (points, badges and leaderboards (PBL)). The approaches don't reflect a true understanding of what makes gamification engaging.

Section two of this paper looks at gamification in higher education. Section three meaningful gamification, motivation theory, fun and what higher education can learn from gamers. Section four discusses the application of gamification to a department of computer science. More specifically it discusses the objectives, target behaviors, the players, fun and deploying the appropriate tools. Section five discusses some of the potential challenges of applying gamification to a CS department and section six discusses our conclusions.

## 2 Gamification in higher education

It has been argued that gamification is a fad and runs the risk of learning becoming a game where people participate only to achieve the game components rather than to improve their skills or knowledge [11]. This can be true if gamification is not done correctly. A good example of this is the conventional model of higher education. Institutions of higher learning are already gamified [12]. Students get points for assignments and exams. These exams translate to grades which affect GPA. GPA gets students on the Dean's list, which is the equivalent of an achievement or badge. When students successfully pass a year of classes they 'level up.' At the end of their matriculation some students get honors, which could be considered the equivalent of another badge or an achievement. One student or a select few make it to the top of the leaderboard and receive class superlatives, such as, valedictorian, salutatorian, summa cum laude, or magna cum laude. This system can lead to some students being highly performance learning oriented and focused on their GPA and resume and not enough on mastering their practical skills and knowledge [13].

Supplementing or changing the conventional model of higher education gamification may represent an opportunity to address the motivation and engagement problems being experienced. One of the main problems with the conventional model of higher education gamification is the fact that 'badges' and 'achievements' are reserved for a select few who demonstrate mastery in a specific way. This form of gamification does not reward the masses.

Researchers continue to implement new gamification research efforts in the pattern of the conventional model of higher education [14][15][16][5][17] [18][19][20][28] [21]. They focus mainly on academic achievement and create new instantiations of points, badges and leaderboards that parallel traditional classroom assessment. Publications regarding higher education gamification research also often lack a theoretical underpinning that can help readers understand the researchers' motivation and the justifications for how their gamification approach is supported by any theory of change. One can say it has been a throw it against the wall and see what sticks approach. This means that findings are difficult to generalize and don't really contribute to the larger body of gamification knowledge. For a literature review of empirical studies on gamification the reader is referred to [4].

Thus far we have only come across two instances of applied gamification [22][5] that reflect a true understanding of what makes games successful and gives some insight into their game dynamics and mechanics. One of them is at RIT [22][23], the other is at Queensland University of Technology Brisbane, Australia [5]. RIT's effort, [22], was designed to give students a more balanced perspective of achievement (academic and social) and took into consideration motivation, engagement and fun. The fact that this gamification effort was carried out in their School of Interactive Games and Media goes to show that other researchers need to have an understanding of the theory and underpinnings of engaging games. One of their major accomplishments is that their approach led to an emergence of peer tutoring sessions that students continued into successive semesters.

# 3 Meaningful gamification

The current emphasis on game components overlooks the true nature and potential of gamification which is to create experiences that users engage with voluntarily. Gamification is a process of creating engagement loops that influence users to perform desired activities [24]. Meaningful gamification can be done without the explicit integration of game components [25]. This viewpoint moves researchers from the shallow perspective of the implementation of game components and puts the focus on the most important aspects of gamification, 'gamefulness' [26]. Gamefulness focuses the system designer on designing, motivation, engagement and fun that lead to change [27][28].

## 3.1 Motivation

Motivation has been shown to increase time on task, direct behavior toward goals, increase effort and persistence as well as affect cognitive processes that impact learning [29]. Characteristics of motivated learners include enthusiasm, focus, persistence and self-regulation, all of which this research seeks to foster or increase.

There are two types of motivation, extrinsic and intrinsic motivation. Extrinsic motivation exists when motivation is aroused by forces outside of an individual. Extrinsic motivators in higher education include grades, scholarships, internships and honors. Intrinsic motivation is aroused from within an individual. Self Determination Theory suggests that there are three intrinsic tendencies that motivate people, autonomy, relatedness, and competence [30]. Maslow's Hierarchy of Needs also supports this theory. It recognizes a person's need for competence and relatedness in the form of self-actualization and belonging respectively [29]. The reader is referred to [31] for an in-depth understanding of how relatedness, competency and autonomy is applied to gamification

Motivation is one of the foremost problems in education [32]. Of the two types of motivators, intrinsic motivators are the more desired. Csikszentmihalyi describes an extreme state of intrinsic motivation and self-regulation called flow [33]. Flow is characterized by a state of complete absorption, focus, and concentration in a challenging activity, to the point that the learner loses track of time and completely ignores other tasks. Flow is only achieved when a task is in the correct balance between not being too easy or too hard and is something that a user is interested in.

Higher education often stifles students' sense of autonomy by not giving them assignments that are socially relevant and by imposing deadlines that are inflexible. Students who are competent in one learning outcome of a class are left languishing, while in the same class they can be penalized for not making a deadline on another learning outcome. In the current structure of higher education there is also little incentive for students to help other students in communities of practice. Faculty take it upon themselves to structure the learning environment without much input from their students. This leads to a learning environment that heavily emphasizes extrinsic motivators.

Research in behavioral economics has shown that extrinsic motivators such as rewards offer short term boosts in activity but can reduce long term intrinsic motivation [34]. In developing and enhancing intrinsic motivation in students, extrinsic motivators should be avoided [35]. The reward and achievement structure of academia is currently totally hinged on extrinsic motivators.

The most useful taxonomy for motivation of students in the context of gamification is Bartle's' player dimension. Bartle categorized players into 4 categories: achievers, explorers, socializers and killers [36]. Achievers enjoy mastering situations and seek out status. They are ambitious, high achieving students, who strive to gain mastery. These students need to be influenced to view mastery beyond solely academic achievement. Explorers enjoy new knowledge and are always looking for a new challenge. They are curious and do not require mastery of material, merely competence. These students should be rewarded for bringing new perspectives to the learning environment. Socializers participate mainly because they enjoy interacting with and being affirmed by other members in the community. They are easily influenced and their standards rise and fall with the standards of the group. These students would benefit most and thrive from communities of practice. These are the students who should be recognized for creating events that improve technical and non-technical aspects of members of the community. Killers enjoy competition. They want to excel and achieve at the expense of other students. They seek to demonstrate superiority and are highly motivated by status and reputation. These are the students who should be targeted to represent their department externally. Their competitiveness can affect the comfort of explorers and socializers. The current dynamics and mechanics of higher education mostly appeal to achievers and killers. A more inclusive and engaging environment should also strive to recognize and reward the other players types.

## 3.2 Defining fun

An understanding of fun contributes to the discussion of engaging learners. Leblanc, [37], describes fun from the perspective of the motivation of the player. He describes 8 types of fun: sensation, fantasy, narrative, challenge, fellowship, discovery, expression and submission. Sensation fun is fun that is pleasing to a player's sense. Fantasy fun is fun that engages escapism and immersion. Narrative fun is fun that unfolds a story. Challenge fun engages the players need to test themselves, overcome and achieve. Fellowship fun engages players in social interaction and cooperation. Fellowship fun does not however encompass competition. Discovery fun is fun that is derived from exploration and learning new things. Expression fun is fun that is derived from expressing creativity. Submission fun is fun derived from gaining accomplishments for tasks that are not cognitively taxing.

## 3.3 Learning from game designers

Characteristics of good games are [38]:

1. The objective or goal of the game is achievable but not too easily.
2. The task is perceived to be fair, i.e. all participants have a similar chance of winning.
3. The stakes for failure are not high.
4. There is sufficient feedback, both positive and negative.
5. There are some elements of chance.

Most gamers would not play a game where they can achieve the objectives on their first attempt, because the lack of challenge does not make them feel competent or self-actualized. Usually as a player develops mastery and increases in level, games get progressively harder. Good game designers ensure that their players can achieve a state of flow. In order to achieve this they allow users the option to control the level of difficulty. This caters to a players need for autonomy while still allowing them to 'win'/feel competent. This can be done in higher education through progressively scaffolded assignments for students who are struggling, or creating 'authentic' projects outside of the classroom for higher achieving students. Higher achieving students would have to earn their way into being members of these projects. Struggling students avoid being overwhelmed and frustrated and high achieving students earn the 'achievement' of the opportunity to work with a team on real world problems that are of interest to them.

Good games reduce the stakes of failure by celebrating learning and reinforcing experiential learning. Formative failure, agency, and choice are seen as critical elements of a true gaming experience [33]. Players are allowed to continue from nearby checkpoints after failure. This ensures that players get frequent feedback on their progress while reducing the stakes of failure. By making the consequences of failure small and integrating elements of chance, game designers allow players to attribute a lack of success to chance and maintain their self-worth. Self-worth, self-concept and affect have been shown to have a complex relationship with motivation and determines whether or not an individual will continue a task [16]. Checkpoints in higher learning are usually summative, measured in semesters in which failure is very hard to recover. The increasing pervasiveness of auto graders and online courses can make it easier for students to get more frequent feedback and feel a sense of accomplishment. Sufficient feedback helps reinforce motivation and gives the player cues on how to interact [42]. Faculty should also consider rewarding more competent students for helping other members of the community.

Good games are designed for collaboration and encourage interaction between players both in game and out of game. In-game chats allow players to point out errors in other players' game play or strategy. Forums, blogs and wikis allow player generated and moderated content to benefit the community as a whole. More experienced players are spotlighted which satiates their sense of competence and increases their sense of relatedness while creating role models for other members of the community. Clear and immediate feedback to students does not necessarily have to come at the cost of faculty and TA's time. Higher education should emulate this type of collaborative environment. A good gamified example of this kind of dynamic at very little long term cost in man hours to the creators is Stack Overflow.

### 3.4 Gamification framework

To address meaningful gamification Werbach, [7], has put forward an iterative user centric six step Gamification Design Framework: 1. Define objectives; 2. Delineate target behaviors; 3. Describe the players**;** 4. Devise the activity cycles; 5. Do not forget the fun**;** 6. Deploy the appropriate tools**.**

The first five steps of this approach address the sociocultural aspect of gamification that is necessary to ensure that it is meaningful. Step 6 is the deployment of technical components that act as reinforcers to facilitate the changes desired in the first five steps.

## 4 Departmental gamification

Gamification can be used to improve the holistic development of a department's students by creating an ecosystem that can be used to influence a department's culture both inside and outside the classroom. Social Cognitive Theory states that social interactions act as response-consequence contingencies that help to model appropriate behavior, beliefs and attitudes [37]. Social interactions, the environment and the cognitive models of members of a community, all have reciprocal relationships on each other and influence the culture of a community. Gamification can be used to guide the enculturation of new students to the CS community while giving feedback to current members of the community about the needs and values of other members. Understanding the current culture of the community and its environment is critical to influencing change through gamification.

Any gamification effort should take into consideration how it may impact a department's learning environment. A department's learning environment consists of the "… physical surroundings, psychosocial or emotional conditions and social or cultural influences" present [29]. Each of these factors plays a role in influencing a student's sense of belonging and their achievement. Research shows that there may be a need to address the fact that CS environmental culture can be more competitive than collaborative [39][40]. Research, [41][42] , also shows that artifacts in the environment can affect a student's sense of belonging. Gamification can be used to enhance inclusivity and diversity. Inclusivity and diversity in this context includes race, gender, areas of study in the field, and achievement (not limited to academic).

This process, like any process that involves humans, needs to be adaptive and nondeterministic. What it means to be a computer scientist is subjective and the field of computer science is continuously changing. Internal and external feedback on the game mechanics and dynamics used are essential to ensure that the process is objective and truly holistic in the development of the student. In order to define the objectives and target behaviors of members of the community, feedback should be sought from all stakeholders of the department as well as members of the education community. We have sought input from, students, alumni, faculty, industry partners, faculty of other programs, the CS education community, educational psychologist and sociologist. One process that we have found useful for understanding and designing our gamification approach is Google Venture's approach to design sprints [43] and the "How might we", HMW, approach [44].

### 4.1 Defining objectives

Through mini design sprints [43][45] the following high level objectives have been identified:

1. Emphasize inclusivity and diversity in the ecosystem.
2. Improve department members' sense of community index [46].
3. Foster mastery learning orientation versus performance learning orientation ,[13], by creating formative 'fail often, fail fast' innovative communities of practice that encourage students to learn through experimentation and trial and error.
4. Create an explicit onboarding system that successfully assimilates students into the CS community.
5. Emphasize the importance of non-technical skills as well as technical skills to success.
6. Encourage students to contribute to the development of academic content as well as mentoring their peers.
7. Create a system of recognition for students that also takes into consideration factors other than solely academic achievement.
8. Highlight role models and their paths to success.
9. Allow students to give immediate feedback to faculty regarding their pedagogy throughout the semester.
10. Create a method of data capture that helps the department to understand how students are spending their time preparing for life after academia (workshops, extracurricular projects, hackathons, networking, groups, internship, etc.).

Every department and their current culture is unique and the objectives for a gamified system should vary.

### 4.2 Delineating target behaviors

Based on the aforementioned objectives a few of the high level target behaviors we have identified for students are as follows:

1. Increased student participation in extracurricular CS related activities.
2. Increased social interaction and networking between students and faculty/alumni/
industry partners.
3. Student initiated and regulated communities of practice.
4. Community generated materials for learning.
5. Student alumni/faculty/industry partnership and mentorship.
6. Community Regulation. Members of the community shall be involved in the assessment of other members.

### 4.3 Describing the players

Each department's has their own unique set of stakeholders. Each department's students/ alumni/industry partners and faculty members will have different demographics, needs and skills to offer. With regards to our students we have done literature reviews, spoken to alumni and to industry partners to see where we are hitting the mark and where we are falling short with regards to preparation for life after graduation. As a Historically Black College and University (HBCU), we have identified threats to belonging [47] in the CS community that affect underrepresented students and have been working to remediate these through interventions and, changing the social and classroom culture of the department.

### 4.4 Devise the activity cycle

Activity cycles consists of two types of loops, progression loops and engagement loops [7]. Progression loops are a series of cycles of growth followed by intermediate stages of mastery. In each cycle the student is learning or acquiring new skills, followed by a period of demonstration of mastery. After the student has demonstrated mastery of content or skill they enter a new cycle in order to master a new skill or activity. The progression loop continues until the student progressively masters all of the cycles. Engagement loops are cycles of motivation, action and feedback that reinforce the user for doing an action and motivates them to continue performing the desired target behaviors. For a thorough review of engagement strategies in games and how they can inform the instructional design of activity cycles the reader is referred to [19][8][48]. To ensure that our game components cater to all of Bartle's player types, we ask, 'How might we make this target behavior appealing to {achievers, explorers, socializers, killers}?' [42].

### 4.5 Not forgetting the fun

In order to ensure that the ecosystem being developed is truly engaging to all our stakeholders, we have been systematically including our stakeholders in design sprints so that we can get their feedback. As we design and develop the dynamics, mechanics and components of the system we ask, 'How might we include {sensation, fantasy, narrative, challenge, fellowship, discovery, expression, submission} fun into this aspect of the ecosystem?' [43]

### 4.6 Deploy the appropriate tools

The specific instantiations of game components used should be different from department to department. Achievements can be virtual in the form of badges, titles, levels or ranks, or they can be tangible in the form of recognition in front of peers, certificates, awards, recognition in the college magazine, unlocking special invitations to industry partner recruiting events, etc. One thing that we would like to highlight is that game lore and narratives play a big part in the engagement process of full video games. Currently our department does not have an explicit method of passing on stories about outstanding achievements of students, alumni, faculty, the department and members of underrepresented groups in CS. Content about these achievements shall be strategically pushed throughout the ecosystem via mobile/web/and interactive displays throughout the department.

### 4.7 Assessing the gamification process

Measuring the effects of gamification can be categorized into two categories, behavioral outcomes and psychological outcomes [4]. Behavioral outcome assessment tracks the delineated targeted outcomes that have been set for the enculturation process. They include soft skills (team work, communication) and professional development (leadership and interviewing), technical skills (requirements analysis, system design, software version control, project management, testing and large scale software development competence), and community interaction. Most of these behavioral outcomes shall be assessed using descriptive and inferential statistics. Psychological outcomes focus on tracking internal changes such as motivation, engagement, self-efficacy, self-regulation [49], co-regulation [50], time management [50], help seeking [50], attitudes toward CS [51], learning goal orientation [13], and sense of community [46]. Research is currently being done into finding other validated psychometric tools that would be appropriate.

## 5 Potential challenges

Several challenges have been identified in implementing meaningful gamification in a CS department. A few are:

1. Not increasing the demand for faculty time.
2. Maintaining an appropriate equilibrium between pursuing technical ability and other aspects of membership of the discipline.
3. Validating and adapting technical ability and membership of the discipline achievements beyond peer review.
4. Designing a system so that rewards do not decrease students' intrinsic motivation.
5. Legal risks [52]
    a. Sweepstake laws if tangible rewards are introduced to system
    b. Privacy - protecting personably identifiable information
    c. Intellectual property concerns regarding content or artifacts generated by community members.
6. Ethical Risks
    a. System is too addictive & causes burn outs.
    b. Peer pressure forces students to participate.
    c. Leaderboards – could result in an over focus on status and may only be effective in the short term [53].
    d. FERPA prohibits access to an educational institution's database of resumes. Could a student's profile be considered a resume?
7. Preventing gaming the System (Cheating and collusion.)
8. Designing flexible engagement and progression loops so that they support flow yet achieve the desired outcome.

# 6 Conclusion

Gamification presents an opportunity to model and guide the enculturation process of computer science undergraduates. It also has the potential to make the matriculation journey more engaging for a broader range of students if done correctly. Understanding the current culture of CS and the readers department is critical to influencing change through gamification. This process, like any process that involves humans, needs to be adaptive and nondeterministic. What it means to be a computer scientist is subjective and the field of computer science is continuously changing. Werbach's iterative design framework offers a good reference point for seeking change through gamification. Meaningful gamification should focus on creating engagement loops that motivate users to perform desired activity and can be done without explicit integration of game components.

An in depth understanding of motivation and fun highlights why some approaches to gamification are not as successful as the researchers would have expected. Both internal and external feedback on the game mechanics and dynamics used are essential to ensure that any gamification process is objective and truly holistic in the development of the student.

# References

[1] A. Begel and B. Simon, "Struggles of New College Graduates in Their First Software Development Job," in *Proceedings of the 39th SIGCSE Technical Symposium on Computer Science Education*, New York, NY, USA, 2008, pp. 226–230.

[2] A. Radermacher, G. Walia, and D. Knudson, "Missed Expectations: Where CS Students Fall Short in the Software Industry," *CrossTalk Mag. - J. Def. Softw. Eng.*, no. Jan/Feb, pp. 4–8, 2015.

[3] C. Loftus, L. Thomas, and C. Zander, "Can Graduating Students Design: Revisited," in *Proceedings of the 42Nd ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2011, pp. 105–110.

[4] J. Hamari, J. Koivisto, and H. Sarsa, "Does Gamification Work?–A Literature Review of Empirical Studies on Gamification," in *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, 2014, pp. 3025–3034.

[5] Z. Fitz-Walter, D. Tjondronegoro, and P. Wyeth, "Orientation Passport: Using Gamification to Engage University Students," in *Proceedings of the 23rd Australian Computer-Human Interaction Conference*, New York, NY, USA, 2011, pp. 122–125.

[6] S. Deterding, M. Sicart, L. Nacke, K. O'Hara, and D. Dixon, "Gamification. Using Game-design Elements in Non-gaming Contexts," in *CHI '11 Extended Abstracts on Human Factors in Computing Systems*, New York, NY, USA, 2011, pp. 2425–2428.

[7] K. Werbach and D. Hunter, *For the Win: How Game Thinking Can Revolutionize Your Business*. Philadelphia: Wharton Digital Press, 2012.

[8] N. Eyal, *Hooked: How to Build Habit-Forming Products*. Portfolio, 2014.

[9] "Can't play, won't play," *Hide&Seek*. [Online]. Available: http://hideandseek.net/2010/10/06/cant-play-wont-play/. [Accessed: 28-Mar-2015].

[10] I. Bogost, "Persuasive Games: Exploitationware." [Online]. Available: http://www.gamasutra.com/view/feature/6366/persuasive-%20games-exploi%20tationware.php/. [Accessed: 28-Mar-2015].

[11] G. Boulet, "Gamification: The Latest Buzzword and the Next Fad," *eLearn*, vol. 2012, no. 12, p. 3, Dec. 2012.

[12] J. J. Lee and J. Hammer, "Gamification in education: What, how, why bother?," *Acad. Exch. Q.*, vol. 15, no. 2, p. 146, 2011.

[13] C. S. Dweck and E. L. Leggett, "A social-cognitive approach to motivation and personality," *Psychol. Rev.*, vol. 95, no. 2, pp. 256–273, 1988.

[14] C. Cheong, F. Cheong, and J. Filippou, "Quick Quiz: A Gamified Approach for Enhancing Learning," 2013.

[15] P. Denny, "The Effect of Virtual Achievements on Student Engagement," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2013, pp. 763–772.

[16] A. Domínguez, J. Saenz-de-Navarrete, L. De-Marcos, L. Fernández-Sanz, C. Pagés, and J.-J. Martínez-Herráiz, "Gamifying learning experiences: Practical implications and outcomes," *Comput. Educ.*, vol. 63, pp. 380–392, 2013.

[17] L. Haaranen, P. Ihantola, L. Hakulinen, and A. Korhonen, "How (Not) to Introduce Badges to Online Exercises," in *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2014, pp. 33–38.

[18] L. Hakulinen, T. Auvinen, and A. Korhonen, "Empirical Study on the Effect of Achievement Badges in TRAKLA2 Online Learning Environment," in *Learning and Teaching in Computing and Engineering (LaTiCE), 2013*, 2013, pp. 47–54.

[19] A. Iosup and D. Epema, "An Experience Report on Using Gamification in Technical Higher Education," in *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2014, pp. 27–32.

[20] C. Li, Z. Dong, R. H. Untch, and M. Chasteen, "Engaging Computer Science Students through Gamification in an Online Social Network Based Collaborative Learning Environment."

[21] L. Sheldon, *The Multiplayer Classroom: Designing Coursework as a Game*, 1 edition. Austrailia ; Boston, Mass: Cengage Learning PTR, 2011.

[22] A. Decker and E. L. Lawley, "Life's a Game and the Game of Life: How Making a Game out of It Can Change Student Behavior," in *Proceeding of the 44th*

*ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2013, pp. 233–238.

[23] E. L. Lawley and A. Phelps, "'You Know You're Going to Fail, Right?': Learning From Design Flaws in Just Press Play at RIT."

[24] K. Werbach, "(Re)Defining Gamification: A Process Approach," in *Persuasive Technology*, A. Spagnolli, L. Chittaro, and L. Gamberini, Eds. Springer International Publishing, 2014, pp. 266–272.

[25] S. Nicholson, "A User-Centered Theoretical Framework for Meaningful Gamification," in *Games+ Learning+ Society 8.0, Madison, WI*, 2012.

[26] S. Deterding, D. Dixon, R. Khaled, and L. Nacke, "From Game Design Elements to Gamefulness: Defining 'Gamification,'" in *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, New York, NY, USA, 2011, pp. 9–15.

[27] S. Deterding, "Gamification: Designing for Motivation," *Interactions*, vol. 19, no. 4, pp. 14–17, Jul. 2012.

[28] J. McGonigal, *Reality Is Broken: Why Games Make Us Better and How They Can Change the World*, Reprint edition. New York: Penguin Books, 2011.

[29] J. E. Ormrod, *Human Learning*, 6 edition. Boston: Pearson, 2011.

[30] R. M. Ryan and E. L. Deci, "Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being.," *Am. Psychol.*, vol. 55, no. 1, p. 68, 2000.

[31] F. Groh, "Gamification: State of the art definition and utilization," *Inst. Media Inform. Ulm Univ.*, vol. 39, 2012.

[32] C. Ames, "Motivation: What teachers need to know," *Teach. Coll. Rec.*, vol. 91, no. 3, pp. 409–421, 1990.

[33] M. Csikszentmihalyi, *Flow: The Psychology of Optimal Experience*, 1ST edition. New York: Harper Perennial Modern Classics, 2008.

[34] D. H. Pink, *Drive: The Surprising Truth About What Motivates Us*. Riverhead, 2011.

[35] E. L. Deci, "Effects of externally mediated rewards on intrinsic motivation.," *J. Pers. Soc. Psychol.*, vol. 18, no. 1, p. 105, 1971.

[36] R. Bartle, "Hearts, clubs, diamonds, spades: players who suit MUDs. 1996," *Game Des. Read. Rules Play Anthol.*, pp. 754–787, 2003.

[37] R. Hunicke, M. LeBlanc, and R. Zubek, "MDA: A formal approach to game design and game research," in *Proceedings of the AAAI Workshop on Challenges in Game AI*, 2004, pp. 04–04.

[38] K. Becker, "Pedagogy in commercial video games," *Games Simul. Online Learn. Res. Dev. Framew.*, pp. 21–47, 2007.

[39] S. Beyer, K. Rynes, and S. Haller, "Deterrents to women taking computer science courses," *IEEE Technol. Soc. Mag.*, vol. 23, no. 1, pp. 21–28, Spring 2004.

[40] L. J. Barker, M. O'Neill, and N. Kazim, "Framing Classroom Climate for Student Learning and Retention in Computer Science," in *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, New York, NY, USA, 2014, pp. 319–324.

[41] S. Cheryan, V. C. Plaut, P. G. Davies, and C. M. Steele, "Ambient belonging: How stereotypical cues impact gender participation in computer science," *J. Pers. Soc. Psychol.*, vol. 97, no. 6, pp. 1045–1060, 2009.

[42] S. Cheryan, A. N. Meltzoff, and S. Kim, "Classrooms matter: The design of virtual classrooms influences gender disparities in computer science classes," *Comput. Educ.*, vol. 57, no. 2, pp. 1825–1835, Sep. 2011.

[43] J. Knapp, "The Design Sprint," *The Design Sprint-Google Ventures*. [Online]. Available: http://www.gv.com/sprint/. [Accessed: 20-Mar-2015].

[44] W. Berger, "The Secret Phrase Top Innovators Use," *Harvard Business Review*. [Online]. Available: https://hbr.org/2012/09/the-secret-phrase-top-innovato. [Accessed: 14-Apr-2015].

[45] V. Paelke and K. Nebe, "Integrating Agile Methods for Mixed Reality Design Space Exploration," in *Proceedings of the 7th ACM Conference on Designing Interactive Systems*, New York, NY, USA, 2008, pp. 240–249.

[46] D. M. Chavis, K. S. Lee, and J. D. Acosta, "The sense of community (SCI) revised: The reliability and validity of the SCI-2," in *2nd international community psychology conference, Lisboa, Portugal*, 2008.

[47] R. Varma, "Making Computer Science Minority-friendly," *Commun. ACM*, vol. 49, no. 2, pp. 129–134, Feb. 2006.

[48] N. Eyal, "Hooked: How to Build Habit Forming Technologies." [Online]. Available: https://www.youtube.com/watch?v=FDtycnZgCfY. [Accessed: 20-Mar-2015].

[49] J. M. Brown, W. R. Miller, and L. A. Lawendowski, "The self-regulation questionnaire.," 1999.

[50] P. R. Pintrich and others, "A manual for the use of the Motivated Strategies for Learning Questionnaire (MSLQ).," 1991.

[51] A. Elliott Tew, B. Dorn, and O. Schneider, "Toward a Validated Computing Attitudes Survey," in *Proceedings of the Ninth Annual International Conference on International Computing Education Research*, New York, NY, USA, 2012, pp. 135–142.

[52] K. Werbach, "Gamification," *Coursera.org*. .

[53] E. D. Mekler, F. Brühlmann, K. Opwis, and A. N. Tuch, "Do Points, Levels and Leaderboards Harm Intrinsic Motivation?: An Empirical Analysis of Common Gamification Elements," in *Proceedings of the First International Conference on Gameful Design, Research, and Applications*, New York, NY, USA, 2013, pp. 66–73.

# An Initial Study of BILOID Humanoid Robot & Beyond

**Yujian Fu, Hiberto Ayala, Jian Fu**
Department of Electric Engineering & Computer Science, School of
Engineering, Technology, and Physical Sciences, Huntsville, AL, USA

**Abstract –** *This paper presents a research study of robotics systems at Alabama A&M University in the past two years. The study has demonstrated several meaningful results that can be further extended in both education, and classroom teaching. In addition, the current study has several potential application areas in the research in formal analysis, image processing and other STEM research and educational areas. The study of robotics systems focus on several different types of current educational and academic versions that are ranged from K8 to K14. We summarize the study by mainly focusing on the BIOLOID GP robots based on the existing preliminary results.*

**Keywords:** Robotics system, Bioloid robot, STEM, education

## 1 Introduction

Using robots in the computer science class and student project is not a new idea [1, 2, 3]. It has been studies in many institutions in the past decades. Moreover, with the fast technology improvement of the computer and electronic science, in the programming and chips, the robotics systems are presented as integrated platforms that provide various supports for interdisciplinary study in the multiple STEM areas. Robotics research study, as the benefit from the fast improvement of technologies, have been dramatically increased in many new cutting edge fields, such as formal methods, computer vision and image processing, artificial intelligent, and so forth.

To improve the current computer science research capabilities at Alabama A&M University (AAMU), reduce the student labor works on other issues, motivate students in the programming learning, and increase students problem-solving skills, robotics project in the classroom teaching and research project for graduate students thesis is the idea option.

This paper presents a research study path of applying robotics systems in the computer science at Alabama A&M University. This study focuses on the application of BILOID GP robots in the past two years in the graduate classroom project, thesis, and undergraduate capstone project. The study of robotics systems in computer science at AAMU falls into several types of academic and educational versions that are ranged from K8 to K14 as well as graduate level.

### 1.1 Robotics in Computer Science Curriculum

Computer science curriculum was updated in the past 3 years to fulfill the fast growth of information technology and meet the student needs in their career. In addition to cyber security concentration, robotics class was added to the computer science curriculum in 2012. Several robotics projects were adopted in the capstone class of computer science at AAMU from 2011 Spring to 2014 Spring semester.

In the graduate level, the LEGO NXT mindstorm kit was the initial platform that was used in the classroom and research project. This LEGO NXT mindstorm is a highly integrated system that assembly the most complicated systems which can be completed by one hour or so. The programming language for LEGO NXT mindstorm is JAVA with the LEGO package (LeJOS). The benefit for students are tremendous, we just enumerate a few. First, students are able to observe how to move the wheel of robot by combining a serials of other actions – these sequence of simple actions can be finished in a small section of program. Secondly, students are dramatically reduced from labor work, such as sensor connection, meld, and battery and other electronic issues. Finally, the main benefit is students are exposed to the real application area of programming, and find out how to write the program and make things work. As the initial try-outs for student projects, the results of LEGO NXT mindstorm is motivating and encouraging. With more challenge research goals and to fulfill the students' needs, we introduce many other robotics platforms.

### 1.2 Research Study in Robotics

The most popular research methodologies in the robotics from current literature review would be artificial intelligence. Artificial intelligence is a traditional research technique and area that use a computer to model and/or replicate the intelligent behavior with minimal human intervention. It has been used in many other areas such as computer vision, e-commerce, medical diagnosis, gaming, just name a few. It is noticed by authors that robotics (esp. LEGO RCX and NXT) were first adopted in many artificial intelligence classes [2, 3]. It is also widely investigated by researchers for the object identification, collision avoidance, path generation, in several other research areas of robotics systems.

Some other research areas step in later. Robotics projects were adopted in the software engineering classes [4, 5, 6], where students were exposed to the system design, implementation. In addition, how to develop a reliable and robust complex systems was addressed in the content. Formal analysis is an infant technique in computer science. Application of formal approach to robotics systems is still quiet young. Our first research project for graduate student on the reconfiguration of robotics systems in LEGO NXT mindstorm was completed in 2011 [7, 8, 9].

This paper is organized as follows. A simple introduction of various types of robotics systems used in computer science at AAMU will be presented in the next section. A detail introduction of BIOLOID with preliminary results will be illustrated in Section 3. The extension to STEM research and educational areas will be discussed in Section 4. Finally, the analysis and conclusion will be presented in Section 5.

# 2 Robotics Systems Used in Computer Science Program at AAMU

In this section, we introduce the several robotics platforms that have been used in computer program at AAMU in the past years. Some student projects developed on these platforms were presented.

## 2.1 LEGO NXT Mindstorm

The LEGO NXT Mindstorm kit is equipped with the intelligent NXT brick, sensors, battery and cables other than all other necessary pieces. The NXT brick is featured with a 32-bit microprocessor, a large matrix display, 4 input and 3 ouptut ports, embedded Bluetooth, USB communication and a speaker. All NXT mindstom kits include 3 interactive servo motors to give you precise control over the robot and ensure that the robot moves with precision – the built-in rotation sensor can measure one degree steps. Among five sensors (2 touch sensors, color sensor, sound sensor and ultrasonic sensor), color sensor has triple functionality that distinguishes black and white, a range of bright and pastel colors, and light settings, as well as can used as a color lamp.

LEGO NXT software is a diagram-based programming platform that well supports K5 to K8 kid's programming learning programming logics conceptually with user-friendly interface. In addition, the third party of LEGO has developed several high level programming platforms that replace the NXT firmware. In our classes, we use LeJOS.

LeJOS – was created in 1999 by a hobby open source project named Jos Solrzano, and then replaced for the NXT bricks. Now LeJOS [10] is a Java operating system which replaces the original firmware of the NXT brick. The consequence of this replacement is that the NXT no longer runs LEGO code from the software, but Java code. The LeJOS package can be included into the IDE like eclipse and NetBeans. Then the brick will be flashed with the LeJOS firmware after compilation. LeJOS package has included all necessary APIs that support the movements of LEGO kit, mostly sensors and servos. Other than this, LeJOS is mainly a pure Java programming language that follows the syntax and semantics of Java. The students can completely enjoy the journey of learning Java while playing with the LeJOS and LEGO kit through the project period.

Since LeJOS is purely a Java program, students can use all object-oriented structure of Java to manipulate the details of sensor or motor and address issues through the program. All abstract tasks like behavior and navigation are offered in several classes. Here is a simple code segment, which illustrates the use of LeJOS in LEGO NXT. The code checks if the front of the bot has an object on the way within specified distance. If yes, rotate 90 degree and returns true. Otherwise, returns false.

```
public static boolean isFrontBlocked(){
    UltrasonicSensor Sonar = new UltrasonicSensor(SensorPort.S2);

    Motor.A.setSpeed(50);
    Motor.A.rotate(90);
    if (Sonar.getDistance() < 30) {

        Motor.A.rotate(-90);
        return true;
    }else{
        Motor.A.rotate(-90);
        return false;
    }

}
```

## 2.2 Arduino

The basic hardware of Arduino consists of a Microcontroller Unit (MCU) based on an 8-bit Atmel AVR Reduced Instruction Set Computer (RISC) microcontroller (e.g. Atmel 2560, Arduino Pro Mini and Uno versions) running at a clock speed up to 8/16MHz, with 32kB flash memory (0.5kB are used by bootloader) and 2kB Static Random Access Memory (SRAM). In addition, it has included on-board, as standard features, 11 digital pins which can be set as either an input or output as well as 7 analog input pins (e.g. Arduino Pro Mini). It is also designed with standard pinout shields, which are used to configure the input and output pins.

Software – The Arduino software platform [11] is designed as open source platform with Java skin on a C-like Programming Language. Arduino provides a plenty of APIs for the interface, control and communications. It supports mouse, keyboard, sensor and serial data manipulate operations. In the control structure, Arduino software follows the Java syntax format in supporting IF-statement, SWITCH-statement, FOR-loop, WHILE-loop. In addition, a bunch of style writing guide is provided for hobbies and developers. A snapshot of Arduino program is shown in Fig. 1.

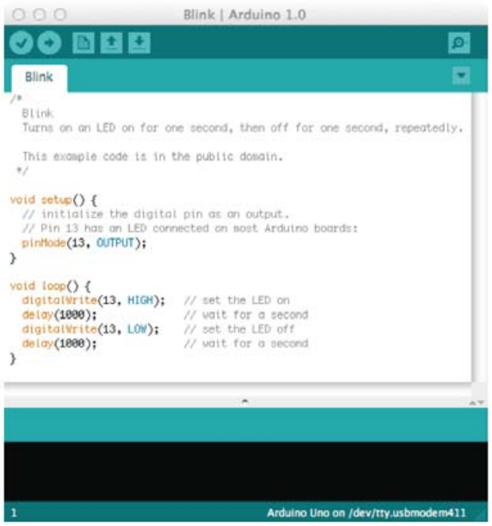



Figure 1. Arduino IDE

## 2.3 Bioloid

Robotis supports multiple versions of Bioloid kit from beginner, comprehensive, expert to premium. In the humanoid style, it supports premium, GP and Darwin. We will introduce the two main version used in AAMU computer science.

### 2.3.1 Bioloid Preminum

Bioloid premium is a medium level robot kit for hobbies, educators and researchers. The central unit of the premium is CM-530 controller [12, 13], which allows the user to easily connect the AX-12A servo motors, the sensors and any communication devices that are needed by the project. Each accessory has a unique ID so that configuration of the system will be unambiguous. Dynamixel AX-12A is an integrated motor with several features, such as position and speed can be controlled in 1024 steps, feedback for angular position, angular velocity, and load torque can be set up in the motion, wiring is with daisy chain connection and supports many dynamixel units with very few resources. Three types of assembly construction were provided. In addition, 23 other types of assembly are supported. The limitation is the actuators in this level does not support enough torque therefore some of the pose cannot be completed in a type of assembly.

### 2.3.2 Bioloid GP

In addition to all the features in premium version, Bioloid GP is an advanced version that supports various types of poses in a more complicate motion.

## 2.4 Other Types of Robotics

The Hovis eco robot [8] is designed with humanoid in mind, just by plastic frame and casing it's one of the most human like robot in the industry. The eco is 16.5 inches height 7.6 inches wide, and weight 4 pounds. It comes with 20 HerkuleX motors from head to foot. One motor in it's head, 7 in the upper body, and 12 in the lower body. The Hovis also has multiple sensors gyro sensors for balance, position sensitive device (PSD) which is used to check position and distance from objects. There is other sensor such as an IR sensor, Bluetooth sensor, touch sensor, led lights. The Hovis eco can be programmed by using with the DR-Sim, which the user can create motion using a 3-D image of the humanoid and set when the motion will activate, or DR- Visual Logic which using a C like language, to drag and drop icons to operate the robot. The more advance options to program the humanoid are Microsoft Robotics developer studio, DR-C, Visual studio, and AVR studio.

## 2.5 Summary

All robots used for research and education work have been proved by studies in many STEM areas. Many literature reviews have demonstrated the successful implementation in use of the robotics to improve student learning outcome in computer science and/or other STEM areas. Currently at AAMU, we have studied four types of robotics systems in both wheeled and bipedal robots in classroom and research project. The most recent work on the Bioloid robotics has shown some primary but exciting results in both research and education [12]. In comparison with LEGO NXT mindstorm, Bioloid (GP) has proved to be an efficient, flexible and reliable robotics system that is able to perform much more complex functionalities and behaviors with the limited labor work on assembly.

In addition, the recent results from our study on LEGO mindstrom and other types of robots, we collected following interesting findings. First, LEGO NXT mindstorm is the most easily assembled robotics system that is able to support multiple Object oriented programming languages such as Java, C++, and C# with the built APIs on the device. Second, Arduido is easy to program on an open source platform and is able to connect multiple accessories with various types of extensions. Thirdly, HOVIS eco is the most friendly humanoid robotics system that is able to play with in a limited programming platform. Finally, Bioloid is the most integrated platform that is able to complete complex motions.

# 3 Current Research Study in BIOLOID

In this section, we introduce the research study and some results using Bioloid robotics systems in past two years.

### 3.1 CM-530 Controller

There are three types of BIOLOID used for STEM areas now – premium is the most popular version. Other than it. BIOLOID GP is the advanced version that is able to perform more complicated tasks. There are some main characteristics embedded in BIOLOID GP robots, including controller, dynamixels and actions that can be performed.

The BIOLOID GP robot is controlled by the CM-530 controller, inside it carries an ARM Cortex STM32F103RE microcontroller. This microcontroller has 64 pin that individually control a part of the robot, such as the input/output and how much voltage is used by the robot. Each pin has a fin number, pin name and a net name that the user must know in order to program them. An example of this for the dynamixel is that it is only controlled by four pins. These pins are used to enable the transmission of data, enable the reception of data, and the other two controls the transmission and reception of data. The names of these pins are ENABLE_TX, ENABLE_RX, DXL_TXD, DXL_RXD, and they are used as a part of the code for the dynamixel to be controlled using embedded C. The BIOLOID GP robot is more versatile by using embedded C instead of the RoboPlus, because every part of the robot can be programmed individually by the programmer. Also it can enable other features that RoboPlus does have such as multi-threading.

### 3.2 Dynamixels

Two types of dynamixels are available for Bioloid GP – AX-12, and AX-18. Dynamixels are complex actuators for robot servo that use daisy-chainable bus connections of 1-3 mbps. The dynamixels receives digital commands of a serial nature, process and send a tatus signal back to the device. Dynamixels work with CM-5, CM-530, or higher level controller well through TTL network protocol of RS485 using USB2Dynamixel adapter. It is found that dynamixels will be automatically shut down if overloaded for 30% to 40% of its torque. There are multiple supporting programming languages including C/C++, Java, Python, C#, Matlab, Labview Visual Studio, etc..

### 3.3 Software – Programming Platform

RoboPlus and embedded C are supported programming platform. RoboPlus is an integrated environment that encapsulates three components – Motion file, Task file, and Manage file – which implements the robot functionalities, configure the programming, and set up each actuator and other accessories in a user friendly interface. Moreover, these three components separate the hardware configuration and parameters from software programming mechanisms. The management file is able to manipulate and reconfigure each individual piece with the reset and debugging capabilities. The relations of the three components in RoboPlus play an integrated and cooperative role on the Bioloid robotics system ranged from as small as a single angle of an actuator turning to as large as the whole system function call. With a very careful and refined setting up of the motion files, and programming on the robotics behavior is becoming a sequence of tasks with an organized structure on the untyped language.

### 3.4 Communication Mechanism – Zigbee

The zig-110a module is used to communicate with the CM 530 controller with the RC remote controller. One of the problems facing this project is using two zig-100 modules to communicate with each other. The zig-100 and zig110a module are both set on the broadcast mode to receive and transmit data but zig-110a module broadcast channel is unchangeable the default value is four. The zig-100 broadcast channel can be changed based on the zigbee pin status, and this used by the RC remote controller. So, in order to have two zig-110a module to communicate with each other is by programming the zig-110a module receive signal terminal (RXD) to receive information from the other zig-110a via the transmit signal terminal (TXD) [13].

### 3.5 Hardware and Externals

BIOLOID GP has a lightweight aluminum frame that comes with eighteen dynamixel motors, eight AX-12 motors for the upper body and ten AX-18 motors for the lower body. It can be controlled with the zigbee wireless module combined with the RC-100A remote controller. This application has an image of the BIOLOID GP humanoid robot with all eighteen dynamixel motors shown by numbers, and those are manipulated by the users to produce the motions of the robot. The motions are numbered from 1 – 255 and each motion can be further manipulated by adding steps to any of the specific motions made by the user. These motions can then be used by RoboPlus task as motion pages that then can be executed by the humanoid robot. The second way the BIOLOID GP humanoid robots can be programmed is in embedded C. When using this form of programming the user must understand the language C and how the BIOLOID GP hardware works.

### 3.6 Current Primary Work & Research Results

BIOLOID GP has a lightweight aluminum frame that comes with eighteen dynamixel motors, Our current BIOLOID GP project aims at completing an autonomous penalty kick by recognizing the incoming ball and performing the kick correspondingly [14]. Initiated in fall 2013 this Bioloid GP humanoid robot is equipped with IR sensor, DMS sensor, and gyro sensor as regular set up. Other than these, an extra IR sensor was added to the right leg to be able to read the incoming object.

The Bioloid GP was first set into its initiation position, from there, the IR sensor was set to a number 50 and if the sensor has a number less than 50. The humanoid robot will kick the ball. Several actions were implemented, due to space limitation, we only attach the code snippet of the task file and shown in the following figures here.

```
START PROGRAM
{
    Initial = TRUE
    Motion Page = 1
    ENDLESS LOOP
    {
```

Figure 2. Initial Position

In this code snippet of Fig. 2, the program is initializing, and the robot going to its default initial position. This will make the robot balance and start the IR sensor, gyro sensor, DMS sensor (motion page number 1). Motion page numbered 1 was configured to the set up of these sensors.

```
        IF ( Button == D )
        {
            Mode = 1
        }
        Buzzer time = Play Melody
        IF ( Mode == 1 )
        {

            Buzzer index = Melody3
            BREAK LOOP
        }
```

Figure 3. Button Press

In the code snippet of Fig 3, after the initialization, the Melody #3 will be chosen and played after you press the button 'D' on the RC. Then a walking step is implemented in a loop of while structure (Fig. 4).

```
    LOOP WHILE ( Motion Page == 78 )
    {
```

Figure 4. Motion Page

Port 2 is recognized the connection port of the Bioloid robot, in this project, in this project, motion page 78 is the walking motion, port is the IR sensor.

```
        IF ( PORT[2] >= 50 && Motion Page == 78 )
        {
            Motion Page = 33
```

Figure 5. IR Sensor

In this snippet of code, if the IR sensor is activated and motion page 78 is running, then the robot should kick the ball, where specification of kick is written in page 33. For the detailed introduction, interest readers refer to the work of [14].

### 3.7 Summary for Bioloid Research Project

BIOLOID GP has a lightweight aluminum frame that comes with eighteen dynamixel motors, and a multitude of sensors that enhance the humaniod compatibility. This project uses both the humanoid compatibility and the sensors to enhance system motion while allow it to kick a soccer ball relative to it's size, fully automatically [14].

Started in fall 2013, we first initialized Biloid robot kicking motion. The basic actions for this robot is to read the coming ball and kick the ball fully autonomatically to another Bioloid robot, which would defend the kick. At the end of year 2013, the kick was realized but without any additional motion. This fundamental motion was done by combining Roboplus task and roboplus motion. The Roboplus motion is used to create motions and roboplus task is used to excutes decision, such as motion pages to execute, sensors used at different situations, and how the robot operates. During past two years the project was improved unpon, by adding a forward motion before the kick, the forward also includes a searching for the ball by turning right and left. Also, the second defensive humanoid robot can use sensors to detect a incoming ball and dectect signal sent from the zigbee wireless by the offensive after it kick with either leg. The defensive robot would defend to the corresponding leg of the offensive robot. This project continues to grow, with enhanced sensors and more intelligent programming.

## 4 Extending to Other STEM Areas

In this section, we introduce the several robotics platforms that has been used in computer program at AAMU in the past years. Some platform with student projects were presented.

In addition to the projects at computer science, authors have noticed that there are large opportunity to extend these project to many other STEM areas such VLSI design in electric engineering and technologies [15], mechanical analysis and dynamic analysis in civil and mechanical engineering [16].

Moreover, humanoid robotics behavior study can be extended to non-STEM areas such psychologies and crime justice.

Finally, we highlighted some new areas in computer science and interdisciplinary research study in the following subsections.

### 4.1 Formal Analysis of Firmware Program Language

After the two year study of BILOID robot programming platform, we find out the RoboPlus is a platform that incoorperate the functionalities of motions in hardware configuration using a user-friendly interface and transparent design. To be able to precisely analyze the semantics of this programming platform for the type of firmware programming language, we introduce formal analysis approach to it [17]. The Lamda is used to analyze untyped programming language, on top of it, we introduce another formal specification language to represent the RoboPlus architecture. Thus, the accurate beavahvior can be captured in both graph and text format.

### 4.2 Kinematic Analysis

Dynamic analysis of humanoid robot is not a new area. Forward and inverse kinematic analysis of robotics system are mainly used to predicate the target and report the position precisely by a mathematical model on the robotics characteristic data, such as length, width, and so on. To be able to calculate the motion accurately, kinematic model is the key of a successful algorithm in the motion and behavior planning of current robotic system. This is most important in the humanoid robotics.

### 4.3 Vision based Object Tracking

Computer vision is a new paradigm in the robotics systems. Due to the avantages of computer vision technique, the object tracking using sequence of frames are investigated and obstacles can be precisely identified To be able to track the continous movement of objects, one of the most popular techniques is the frame based vision tracking on the objects. The frames were captured and analyzed by a high performance algorithm, feedback to be sent to controller. The calculation of the feedback will be used to decide the current object type, position and other charateristics.

### 4.4 Cyber Security in Robotics System

Current malware detection and defense of robotics system is paid more and more attention by researchers with the increased number of malware identification of cyber physical systems and robotics systems [18]. The research areas in this cyber security issues of robotics systems fall in three groups : a) Identification of cyber security and attacks of robotics systems. Robotics systems are a highly integrated multiple disciplinary systems that incorperate with objects, electrics, and paltforms. This indicates that there are more opportunites or security holes for malicous behaviors and attacks. B) Detection of malwares of robotics systems. There are several issues of this area, such as how to detect the malwares of robotics, is the traditional approach still working on the malwares of robotics systems. C) Analyzing malicous behaviors and cyber attacks of robotics systems. Some of the traditional approaches are still working on this area. Some may not work for particular malware attacks. For instance, PLC attacks on the microprocessor may be transferred and amplified to dynamixels and other intellctural accessories of the robotics so that partial control of robot is possible.

## 5 Conclusions

This paper presented the continuous study of using robotics projects at computer science program in Alabama A&M University. Various types of robotics systems were discussed with interested student project study. The work focuses on the BIOLOID GP robots as well as the other types of robotics platforms. This work has demonstrated a continuous efforts that use the robotics in both research and classroom teaching.

The successful implementation of Bioloid robotics system can be extended to agent oriented structure as well as heterogonous robotics control architecture, in which many researchers started working on them. We will continuously work on these areas and seek for the most enthusiastic results in the future.

## 6 Acknowledgement

The robotics project has been supported by title III project under the grant No. DAAD17-02-C-0113. In addition, we appreciate the support of Dean’s Award 2013 of Lucky Martin. Finally, this project is partially supported by Department of Air Force grant FA00056784321.

## 7 References

[1] L A. Stein. “Rethingking cs101: Or, how robots revolutionize introductory computer programming.” Computer Science Education. 1996

[2] E Sklar, S Parsons, and M. Q. Azhar. "Robotics across the curriculum". In AAI Spring Symposium on Robots and Robot Venues: Resources for AI Education. AAAI Press.

[3] Deepak Kumar, Doug Blank, Tucker Balch, Keith O'Hara, Mark Guzdial, and Stewart Tansley. "Engaging computing students with AI and Robotics". In Proceedings of AAI Spring Symposium on Using AI to Motivate Greater Participation in CS, March 2008, AAAI Press.

[4] Gobel S., Jubeh R., Raesch S. L., "A robotics environment for software engineering courses." In Proceedings of 25th AAAI conference on Artificial Intelligence, San Francisco, California, Aug 7-11, 2011, pp. 1874-1877.

[5] Jiwon Shin, Andrey Rusakov, and Bertrand Meyer, "Concurrent Software Engineering and Robotics Education". In Proceedings of International Conference on Software Engineering, 2015.

[6] Davide Brugali and E. Prassler, "Software engineering for robotics," Special Issue on IEEE Robot. Automatic Mag., vol. 16, no. 1, Mar. 2009.

[7] H. Jasti, Y. Fu, "Modeling and Analyzing Web-based Information Systems." In Proceedings of International Conference of Software Engineering and Practice, Las Vegas, NV, Jul 12—15, 2010

[8] Andrew B. Williams, "The quality Impact of Using Lego Mindstorms Robots to Teach Computer Engineering", IEEE Transactions On Education, vol.49, no.1, February 2006

[9] Raffaele Grandi, Riccardo Falconi, and Claudio Melchiorri, "Robotic Competitions: Teaching Robotics and Real-time Programming with LEGO Mindstorms". In Proceedings of 19th World Congress, The International Federation of Automatic Control Cape Town, South Africa. Aug 24-29, 2014

[10] Lew M. W., Horton T. B., Sherrif M. S. "Using LEGO Mindstorms NXT and LEJOS in an advanced software engineering course." In proceedings of IEEE Conference on Software Engineering and Training, pp. 121 – 128.

[11] Arduino Projects. Retrieved From: http://www.adafruit.com/

[12] Chi N. Thai and M. Paulishen, "Using Robotic Bioloid Systems for Instructional Robotics," IEEE SoutheastCon 2011, Nashville, 17—20 March 2011, pp. 300 – 306.

[13] Bioloid Robot E-Manual. Retrieved From: http://support.robotis.com/en/. May, 2015.

[14] Hiberto Ayala, Yujian Fu, "An Automatic Penalty Kick Using RoboPlus.", In Proceedings of IEEE SoutheastCon 2014, Lexington, KY, 14—16, March, 2014.

[15] Andrew Adamatzkey, Paolo Arena. "Reaction-Diffusion Navigation Robot Control: From Chemical to VLSI Analogic Processors." IEEE transactions on Circuits and Systems I, vol 51, no. 5, May 2004.

[16] Bruckmann, T., Mikelsons, L., and Hiller, M. (January 9-11, 2008b). A design-to-task approach for wire robots. In Kecskeméthy, A., editor, Conference on Interdisciplinary Applications of Kinematics 2008, Lima, Peru.

[17] Yujian Fu, M. Drabo, "Formal Analysis of Humanoid Robot", In Proceedings of Embedded System Analysis, 2014.

[18] T. Bonaci, and J. J. Chizeck, "On potential security threats against rescue robotic systems." In Proceedings of IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR'12). Nov 5—8, 2012.

# Learning Hardware Design by implementing student's Video-Game on a FPGA

**M. Sanchez-Elez**[1] **and S. Roman**[1]
[1]Deprt. Arquitectura de Computadores y Automatica, Universidad Complutense de Madrid, Spain

**Abstract** - *This paper presents a methodology to teach the design of complex hardware systems to Computer Engineering students through CAD tools which allow a rapid prototyping and manufacturing of complex systems. The design of such systems requires a broad scope of knowledge and abilities, both for the students and for the teachers. Challenging each student to implement his/her favorite video-game on a FPGA may be an effective method to help them to acquire the necessary skills to design hardware systems with CAD tools. Each student chooses his/her own video-game as the final course project. We have observed that this enhances the motivation required to face the difficulties of handling complex design concepts together with the use of any CAD tools. This yields a more satisfactory experience for students and an increase in the skills gained. An improvement in academic results is also observed.*

**Keywords:** Logic Design, Hardware Description Language, Reconfigurable Hardware, Education

## 1 Introduction

The students who are following any of the different computer grades must deal with computer systems from design to implementation, as they will design computing systems and computing components of products, develop and test their prototypes, and implement them to market. We agree with other [1], [2] and [3] who state that these courses are very important for undergraduate curricula in Computer Science and Electrical and Computer Engineering. These graduates need to manage concepts from many different fields.

The goal is that the students may correctly design a complex algorithm into hardware with the help of any of the existing computer-aided design tools [4], [5], [6] and [7]. This means, they should be able to develop the skills that allow an engineer to rapidly generate hardware descriptio language code (HDL) that a CAD tool synthesizes into the desired hardware, as pointed by [8].

The students have to deal with parallel execution, signal delay, synchronization, etc, and we have observed that it is tough for them to assimilate these concepts. An additional difficulty is that they need to learn a Hardware Description Language in order to simulate and synthesize their designs. On top of this, the synthesis process is platform dependent. This paper describes an experiment carried out along several years where we investigate if the development of a complex video-game project, as an alternative to practical examples in the form of guided lab exercises, may be successfully used to teach hardware design. We do not create a video-game focused on explain hardware design concepts, we suggest to students to design a video game, one that they like best.

Our experience is that from the students` point of view, guided lab exercises seem to be a just a black box with data to collect, which is an approach that is far from real design problems. We suggest that a project based learning (PBL) methodology is more suitable for these courses as it creates a course dynamics that is closer to a professional environment [9].

This paper presents the PBL methodology we use in our courses to fulfill both technical and motivational objectives. The implementation by the students of a complex video-game in hardware not only meets technical design goals, but also highly motivates them in comparison to other type of projects, which is a key aspect in the learning process [10]. Designing and implementing their own video-game as the course project is a key aspect of the methodology, as the students become highly motivated, and therefore more involved and more autonomous in the HW design process. The creation of their own video-game is a challenge related to a reality that is well known for them compared to other possible hardware projects. Most students find it a little hard in the beginning because they have to handle real problems present in hardware design. As their motivation and self-confidence grow along the course, only a minority give up. By the time they are designing their own video-game they become very involved in the course and work really hard. They are autonomous enough to be searching cores that improve their original designs or to be sophisticating the functionalities of the original proposed video-game. This means that the development of their own video-game also helps develop a stimulating and important non-specific cross-skill for engineers as is creativity.

The rest of this paper is organized as follows: Section 2 outlines the educational difficulties present in System Design together with different approaches already pub-lished. Section 3 describes the methodology implemented. Section 4 presents details of the last guided laboratory exercise in which students have to make use of all the previous concepts, before implementing their own video game. Section 5 shows experimental results obtained from students grades statistics and several questionnaires.

## 2 Related Work

In this section the challenges of applying PBL in hardware system design are discussed and related both to already developed educational practices as well as to the knowledge and skills that the EDA industry expects from graduates.

EDA industry has evolved towards the use of CAD tools which allow a rapid prototyping and manufacturing of very complex designs. Hardware design with present software tools do not require such a deep understanding of internal hardware functionalities as it was before [11]. The actual EDA industry requires the management of a wide knowledge in many new areas such as, Hardware Design Languages (HDLs), FPGA-based CAD tools, use of IP cores, soft cores, advanced synchronization techniques, design of virtual systems, etc. [12].

The students who follow these subjects are on their final course, that is, they are about to graduate and start working. Hence, a PBL methodology is convenient for the course development as this is close enough to a real EDA project. The development of a video-game as the contents of the PBL approach is technically as complex as other PBL proposals in this field.

There are several papers [13], [14] which use games as part of the engineering education. These proposals replace the traditional lecture-based passive learning methodology with an active multi-sensory experiential learning methodology. We would like to emphasize that this is not what we are suggesting in this paper. The authors mentioned are using a game as a simulation environment, while we are focusing on the hardware implementation of a video game in a FPGA, in the place of traditional circuit design laboratory exercises.

Most published works about learning methodologies using game projects are inspired by software courses[15], [16]. In these works games are integrated in three ways: traditional exercises can be replaced by games motivating the students to put extra effort in doing the exercises [17], [18]; games can be used within a traditional classroom lecture to improve the participation through knowledge-based multiplayer games [19]; or game development projects can be used in computer science or software engineering courses to learn specific software skills [20].

The last paper mentioned presents a methodology that is focused on the need to engage students in the course, which is also our purpose in this work, as well as to help students face the real problems of a software complex project. Developing a complex game design as a useful educative tool is quite clear for a software design course, but it is not so obvious for hardware design courses. Therefore the convenience of a paper such as this, in which we show that complex video-game developing is as useful in hardware design courses as it is for software design courses.

Some author who also deal with the challenges of teaching hardware design today still hold a traditional point of view of hardware design which they apply to a new hardware target (FPGA), such as [21], or develop complex concepts by using an FPGA-based tool as the main support[22]. In addition, most digital design texts dealing with HDLs [23], [24] present a classical view of digital design concepts merely translated into HDL. They commonly propose the structural design of a simple microcontroller as final project, which adds no significant understanding to the former schematic approach.

The proposal of [25] is also interesting, although these authors make separate simulation and synthesis projects. These are complementary aspects of a whole design process and should be analyzed together from the beginning of the course, since a correct simulation does not guarantee a correct implementation on the target FPGA.

Other authors have shifted from the traditional methods and dedicate half of the semester to implementing a complex final project. Such is the case of [12, 26], in which a video-game is a possible choice for the students. The main difference with the methodology presented in this paper is that in our case the design of their own video-game is the baseline of the course development, being in the mind of the students that the final goal is to create it, which helps maintain an interest and deep understanding from the beginning until the end.

The work presented in [27] uses PBL in a very attractive way, although they use VIP technology, which requires sophisticated laboratory equipment.

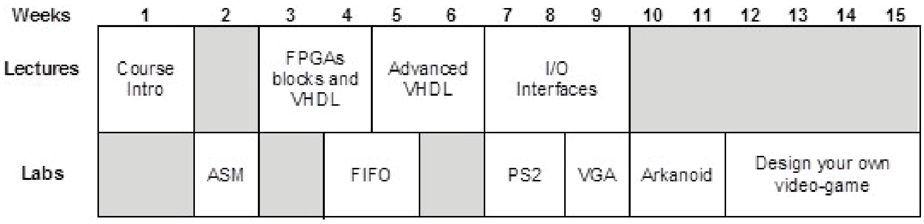



Fig. 1. Curriculum for Advanced Logic Design

Some authors use video-games as the baseline of the course [28], [29] and [30], but in these cases the video-game is not the project itself, it is a tool created by the teachers to help students acquire concepts. These games follow the drill-and-practice model, emphasizing rote memorization and failing to capture the interest of students [31]. The most common approach implies that problems are embedded in a game narrative, and solving them is part of the game [32]. They have turned the whole learning setting into a game, on the idea that playing and solving problems share many learning features. However, in the methodology explained in this paper the problem to be solved is implementing the game itself, which has to be developed by the students.

## 3 Methodology

The methodology assumes that the students are already familiar with: laws of Boolean algebra; binary number representation; basic combinational logic design, basic gates, multi-level digital design; combinational building blocks; basic sequential building block and Finite State

Machine (FSM) design. A basic knowledge of VHDL is also desirable, so a 4 - 6 hours VHDL tutorial [33] is given to the students, regardless of whether or not they have studied it in previous courses.

The key focus of the methodology is on creating the expectation that at the end of the course, once they have mastered certain design challenges, the student are ready to implement their favorite console video-game on a FPGA. All the design created along the course may be used to create the video-game. The project is developed using VHDL as design language, Spartan 3 [34] as prototyping platform and Xilinx ISE [4] as CAD tool.

The syllabus of the course is the following:

1. Course Intro: Introduction to design automation of digital systems. Working framework description: the Spartan.
2. FPGAs blocks and VHDL:
   a. Configurable Logic Block Architecture
   b. Other configurable HW: Block RAMs
   c. Spartan 3 Family
   d. Prototyping boards and the Xilinx ISE Design Suite.
3. Advanced VHDL
   a. Register Transfer (RT) level specification of digital systems using VHDL.
   b. Logic - RT level design techniques.
   c. Logic - RT level analysis of digital systems.
   d. Synthesis for FPGAs.
4. I-O Interfaces
   a. The XESS boards [35]
   b. PS2 Key-board hardware
   c. VGA controller
   d. Music generation

The Fig. 1 describes in a week scheme the syllabus of the course.

The cornerstones of this methodology are the following:

1. At the beginning of the course the instructors present an overall guideline of the theoretical concepts required for the implementation of different hardware modules, all of which may be combined to create any classical video-game.
2. The different hardware modules that the students develop have been specifically designed to present increasing challenges in order to capacitate them to finally implement their own video-game. These modules are based on the develop of different Algorithmic Sate Machine (ASM):
   a. ASM with counters: the teacher provides a guideline of the VHDL code for a synthesizable ASM and a basic counter. The students have to modify this code to implement a traffic light system.
   b. Use of the FPGA internal memories: the VHDL code for a RAM is given together with the analysis of its synthesis process into a BRAM. Students are requested to reuse this code to implement a FIFO.
   c. Input interfaces (PS2 Keyboard): the teacher gives a theoretical introduction on hardware interfaces and details on the PS2. Students are requested to use a given VHDL shift-register to check several key scancodes.
   d. Output interfaces (VGA): a VHDL VGA controller is given so that students may paint several geometric figures on a PC screen.
   e. Guided implementation of an easy video-game Fig. 2. The students implement an Arkanoid video-game completely guied by

the instructors.

3. Midterm group discussion: students are requested to suggest a video-game that may be implemented with the hardware modules developed along the course. This should be presented as a realistic project, that is, they have to evaluate the difficulty of the game by identifying the necessary hardware elements, the amount of reusable code and deadlines. In some cases the teacher addresses the students to either simplify or increase the complexity of their projects, in order to meet the educative goals and deadlines of the course.

The implementation of any video-game choses by the students fulfill the common topics in advanced logic design courses: model, simulate, synthesize and optimize complex subsystems, using in most cases a register transfer level design (RTL). Therefore, for the students who have successfully passed the course the acquired skills are:

1. Correct implementation of flip-flops and registers: when the students attempt to paint different figures and colors on the VGA screen, if their if-else VHDL code is not correct the RGB signal is latched and the colors are not the expected.
2. Correct implementation of a ASM: when the students create the ASM for the moving ball, they learn that the ball does not correctly bounce if the tool has not inferred the ASM associated.
3. Understand the differences between the simulation and the implementation of their designs.
4. Handle several clock domains: the design of the video-game implies using different clock signals for: the VGA controller, the movement of the ball, the video-game controller and the keyboard.
5. I/O management.
6. Use of other partners IP cores: the students feel motivated to enhance their games with new functionalities; as they have to comply with deadlines they are encouraged to make use of existing IP cores such as sound controller, screen backgrounds … and/or reuse modules designed by their classmates.
7. Be aware of the amount of area used by their designs: the students feel inclined to make complex visual designs, which result in a high use of FPGA resources. Thus, the synthesis process becomes slow and in some cases the complete design does not fit the FPGA.
8. Develop self-confidence as hardware designers: once the game is completed, the students themselves realize that they have acquired the skills for hardware design and this motivates them to continue learning.

# 4 Case Study

The easy video-game implementation for the students as first project in order to train for develop their own video-game is the arkanoid. One of the most successful video-game to be implemented in a FPGA is the classical arkanoid, as it is very popular among the students and therefore the majority of them easily understand its operation. In addition, the game is more sophisticated than a tennis game, which highly motivates students as it poses the challenge to implement a fairly complex video-game.

At this point of the course, the students are familiar with the CAD tool and they have implemented the hardware modules described above (BRAMs, PS2 interface, VGA controller) and the instructors give them seven steps they need to follow to correctly implement this game:

1. Paint the video-game edges on the screen, this means to paint a square.
2. Implement a four-state machine in VHDL to emulate the ball movement. The ball bounces off the edges at a 45º angle. The instructor gives the students a VHDL template to create this ASM (Algorithm 1). PX and PY stand for the ball positions on the screen. Each time that the VGA horizontal and vertical counters equal the ball position, the rgb signal value has to be equal to the color of the ball. BallClock stands for the speed of the ball (this clock is slower than the refresh rate of the screen). However, the clock for the ASM is the FPGA main clock.
3. Create the slide-bar. What size is it? How does it move? The students have to decide how many pixels the bar moves with each keystroke.
4. Modify the original ball-ASM in order to emulate the ball bouncing off the slide-bar.
5. Create a matrix of different color rectangles using a BRAM; this could be done as an extension of the VGA controller lab session.
6. Erase color rectangles. Create a VHDL code that changes the color of a rectangle, stored in a BRAM, to background color each time the ball reaches a rectangle.
7. The ball has to bounce off the rectangles if they are not already deleted. When the ball bounces, the rectangle is erased.

By following these seven steps, the students gradually approach the implementation of the arkanoid video-game in an easy.

The students develop the different hardware components required for the final design by means of progressive laboratory exercises. IP cores are also integrated in the project for some parts of the design. This integration of IP cores and gradually developed parts makes the process accessible to all students in a reasonable lapse of time and is also closer to a real development of a complex commercial system.

For instance, the keyboard controller module is developed during week 7 of the course as a practical exercise which consists of implementing a 22-bit shift register, and a with-select VHDL structure that determines which key has been pressed. The VGA controller is developed during week 8 of

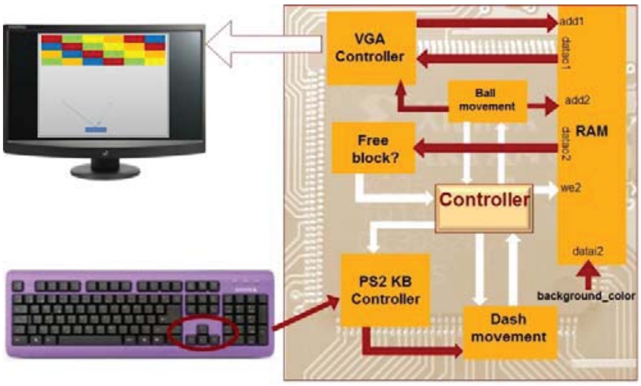



Fig. 2. Block Diagram Scheme for Arkanoid.

the course. The VHDL code of the controller itself is given to them and their task is to assign values to the rgb signal in order to display a relatively complicated geometrical figure on the screen.

As a result, when students are faced with the challenge

```
BallPosition: process(BallClock)
begin
    if BallClock'event and BallClock = '1' then
        case BallState is
            when XpositiveYpositive =>
                px<=px+1;
                py<=py+1;
            when XpositiveYnegative =>
                px<=px+1;
                py<=py-1;
              …
        end case;
    end if;
end process;

BallFSM: process(px, py)
begin
    case BallState is
        when XpositiveYpositive =>
            if px > px_max_edge then
                NextBallState<= XnegativeYpositive;
            elsif py > py_max_edge then
                NextBallState<= XpositiveYnegaitive;
            else
                NextBallState<= XpositiveYpositive;
            end if;
        when XpositiveYnegative =>
        …
        end case;
end process;
```

Algorithm 1. VHDL Template for ball movement

to implement the Arkanoid game, they have already developed the skills and modules they need for the task are may rely upon these to face new challenges such as the management of memory contents and its representation on the screen.

Each rectangle is linked to a different memory word so they need to learn how translate pixels into memory addresses by concatenating the x and y parameters of the pixel. By eliminating the two least significant bits of y and the four least of x, a single memory address may be linked to a 16 x 4 rectangle. A double port memory is required so that it may be read both from the VGA controller as from the game controller.

The game controller also has to translate the ball position into memory addresses. The ball position (px, py) is used to calculate if the ball is encountering a rectangle. If the memory

TABLE 3
Student's Perceptions and Opinions of the Course

content associated to the rectangle is that of the screen, then the ball continues its movement. However, if the memory content is one of the possible colors of the rectangles, the ball has to bounce and the color stored at that memory location has to be changed into the screen color (the rectangle disappears).

Students are taught how this memory functionality may be implemented with the Block RAM modules in the FPGA with a read port connected to the rgb signal and another read/write port associated to the ball movement.

## 5 Results

We started using this methodology on academic year 2010-2011. The courses are held at Universidad Complutense de Madrid (www.ucm.es/english). It is an elective subject for students in their last undergraduate course, with an average of 20 students per class.

TABLE 1
GRADE OBTAINED BY THE STUDENTS

| Grade/Year | 2009/10 | 2010/11 | 2011/12 | 2012/13 |
|---|---|---|---|---|
| A+ | 3% | 7 % | 17 % | 13 % |
| A | 4% | 23 % | 22 % | 40 % |
| B and C | 23% | 40 % | 50 % | 40 % |
| D | 67% | 23 % | 0 % | 0 % |
| E and F | 3% | 7 % | 11 % | 7 % |

The results obtained by the students are shown in (Table 1). Column 2009/10 presents the results obtained when the PBL approach was not centered in the development of a video-game by the students. The other three columns follow the PBL approach with a video-game as the target implementation. Along these last three years we have observed that, on average, 90% of students successfully complete the course. It is also remarkable that more than 30% of the students achieved an A or A+ grade. The good results obtained are explained by taking into account that each school year, the instructors have been refining the practical exercises and the whole methodology. We have also seen increasing motivation from the beginning of the semester in the students enrolling this course along these years. The explanation for this is that a majority choose this course because their fellow students, who have already done it, recommend it to them (we have also noticed an increase in the number of students enrolling the course).

TABLE 2
RESULTS AT THE UNIVERSITY QUALITY ASSURANCE IN HIGHER EDUCATION PROGRAM

| | Score (20010/11) | Score (2012/13) |
|---|---|---|
| Methodology and Scheduling (max 30) | 24.9 | 21.8 |
| Course Development (max 30) | 26.4 | 27.3 |
| Results (max 40) | 35.3 | 34.2 |
| Total (max 100) | 86.6 | 83.3 |

TABLE 3
STUDENT'S PERCEPTIONS AND OPINIONS OF THE COURSE

| Which of these options reflects your perception of the course? | |
|---|---|
| I have been pleasantly surprised | 33 % |
| I liked it | 57 % |
| I found it boring | 10 % |
| I feel disappointed | 0 % |
| The organization and development of the practical sessions | |
| Was interesting and strengthened my interest in HW design | 48 % |
| Was interesting but required a lot of effort | 27 % |
| I managed to comply with the course goals by investing a lot of effort | 5 % |
| I think this methodology demands too much effort | 0 % |
| My perception about the design skills acquired is that | |
| I feel capable of designing any other complex system | 38 % |
| I feel capable of designing systems of a similar nature | 52 % |
| I feel capable of designing systems of a lower complexity | 10 % |
| I still do not feel capable of facing the design of a complex system without assistance | 0% |
| I feel that choosing this course for my curriculum | |
| Is not significant | 0 % |
| Was interesting but I do not think will help me find a job | 14 % |
| Will provide more professional opportunities for me | 62 % |
| Will help me find the type of job I am interested in | 24 % |

Moreover, during academic year 2010-2011 and 2012-2013 we enrolled this course in a university program meant to evaluate teaching performance. Within this program, the results of teaching activities are translated into terms of progress in student learning and in the assessment expressed as perceptions or opinions by students, graduates, academic leaders and the academic staff. This evaluation is coordinated within a program of the Spanish Agency in Quality Assurance in Higher Education, ANECA. This agency is a full member of European Association for Quality Assurance in Higher Education (ENQA) and a full member of the International Network for Quality Assurance Agencies in Higher Education (INQAAHE). The results obtained for this course in this program are shown in Table 2. This evaluation has special mention for being among the 5% top rated of Universidad Complutense de Madrid. Additionally, the instructors of the 2011 - 2012 course have carried out a specific test about students' perceptions and opinions. The results are shown in

(Table 3). This test shows that students are satisfied with the course and feel confident on their skills for designing complex hardware, although for some of them the effort required was significant. Most of them realize that designing hardware systems is a realistic professional path to be considered, a perception that is confirmed when they meet colleagues who are already working in companies that use the same CAD tools and similar hardware targets.

Examples of video-games implementd by the students are:

- Pac-Man (Fig. 3)
- Mario Bros
- Tetris (just four figures)
- Piano learning
- Tron video-game
- Flappy Bird
- Space Invaders

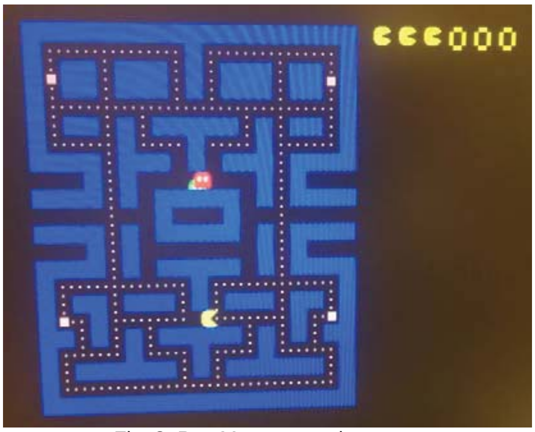


Fig. 3. Pac-Man screenshot.

## 6 Conclusion

The evolution of hardware design processes in EDA industry requires that the former approach for the teaching of hardware design be revised. The use of CAD tools, which are platform dependent, together with the complexity of the hardware design concepts involved make both the teaching and learning quite a hard task.

In this respect we have found many interesting approaches based on the development of a course project similar to real industry projects. However, these types of projects are still too far from the students realities. Implementing a complex video-game on a FPGA triggers students' motivation. The methodology presented has been designed to channel the students' energy through a milestones journey which leads to the video-game implementation. At the end of the journey they have acquired the hardware design skills required for engineering industry today. The results of the anonymous questionnaires from our students support our hypothesis as well.

Moreover, the final outcome is so attractive for them, and the motivation to round off the video-game becomes so strong that many students exceed the undergraduate course goals. This explains why the three years that the video-game has been used as a course project there is an increase in the B, A and A+ marks.

To sum up, it is remarkable that in comparison to the classical course project, the percentage of students that invest the minimum effort to pass the subject (D mark) has decreased and even disappeared. This clearly supports our hypothesis that the motivational aspect of the course project is a key factor in the success of a PBL methodology. Moreover, this methodology is economically sustainable as only a low-priced Xilinx's Spartan 3 platform is required.

# 7 References

[1] C. Harrison and P. Jones, "Experiences with FPGA teaching," in The Teaching of Digital Systems (Digest No. 1998/409), IEE Colloquium on, 1998, pp. 3/1-3/4.
[2] A. McGettrick, M. D. Theys, D. L. Soldan, and P. K. Srimani, "Computer engineering curriculum in the new millennium," Education, IEEE Transactions on, vol. 46, pp. 456-462, 2003.
[3] U. Reinsalu, A. Arhipov, T. Evartson, and P. Ellervee, "HDL-s for Students with Different Background," in Microelectronic Systems Education, 2007. MSE'07. IEEE International Conference on, 2007, pp. 69-70.
[4] "ISE Design Suite Software Manuals and Help " Xilinx Co.2012.
[5] "Quartus II HandBook " Altera2012.
[6] "FPGA based prototyping " Synopsys2012.
[7] "ModelSim VHDL Simulation Tutorial," Mentor Graphics2012.
[8] P. P. Chu, RTL hardware design using VHDL: coding for efficiency, portability, and scalability: Wiley. com, 2006.
[9] A. Yadav, D. Subedi, M. A. Lundeberg, and C. F. Bunting, "Problem-based Learning: Influence on Students' Learning in an Electrical Engineering Course," Journal of Engineering Education, vol. 100, pp. 253-280, 2011.
[10] K. Higley and C. Marianno, "Making engineering education fun," Journal of Engineering Education, vol. 90, pp. 105-107, 2001.
[11] D. D. Gajski and J. Kleinsmith, Principles of digital design vol. 42: Prentice Hall New York, 1997.
[12] V. Sklyarov and I. Skliarova, "Teaching reconfigurable systems: methods, tools, tutorials, and projects," Education, IEEE Transactions on, vol. 48, pp. 290-300, 2005.
[13] A. A. Deshpande and S. H. Huang, "Simulation games in engineering education: A state-of-the-art review," Computer Applications in Engineering Education, vol. 19, pp. 399-410, 2011.
[14] V. Carbonell, C. Romero, E. Martínez, and M. Flórez, "Interactive simulations as teaching tools for engineering mechanics courses," European Journal of Physics, vol. 34, p. 991, 2013.
[15] A. I. Wang, "Extensive evaluation of using a game project in a software architecture course," ACM Transactions on Computing Education (TOCE), vol. 11, p. 5, 2011.
[16] T. Smith, K. M. Cooper, and C. S. Longstreet, "Software engineering senior design course: experiences with agile game development in a capstone project," in Proceedings of the 1st International Workshop on Games and Software Engineering, 2011, pp. 9-12.
[17] B. A. Foss and T. EIKAAS, "Game Play in Engineering EducationĐ Concept and Experimental Results," 2006.
[18] G. Sindre, L. Natvig, and M. Jahre, "Experimental validation of the learning effect for a pedagogical game on computer fundamentals," Education, IEEE Transactions on, vol. 52, pp. 10-18, 2009.
[19] B. Schmitz, R. Klemke, and M. Specht, "Effects of mobile gaming patterns on learning outcomes: a literature review," International Journal of Technology Enhanced Learning, vol. 4, pp. 345-358, 2012.
[20] A. I. Wang and B. Wu, "Using game development to teach software architecture," International Journal of Computer Games Technology, vol. 2011, p. 4, 2011.
[21] I. Vélez and J. F. Sevillano, "A course to train digital hardware designers for industry," Education, IEEE Transactions on, vol. 50, pp. 236-243, 2007.
[22] E. Guzmán-Ramírez and I. Garcia, "Using the project-based learning approach for incorporating an FPGA-based integrated hardware/software tool for implementing and evaluating image processing algorithms into graduate level courses," Computer Applications in Engineering Education, 2012.
[23] P. J. Ashenden, The designer's guide to VHDL vol. 3: Morgan Kaufmann, 2010.
[24] C. H. Roth, Digital systems design using VHDL vol. 20: PWS publishing company, 1998.
[25] A. J. Araujo and J. C. Alves, "A project based methodology to teach a course on advanced digital systems design," WSEAS Transactions on Advances in Engineering Education, vol. 5, pp. 437-446, 2008.
[26] F. Machado, S. Borromeo, and N. Malpica, "Project based learning experience in VHDL digital electronic circuit design," in Microelectronic Systems Education, 2009. MSE'09. IEEE International Conference on, 2009, pp. 49-52.
[27] C.-G. Haba and L. Breniuc, "Teaching FPGA embedded system design by course correlation and with the help of VIP systems," in Proceedings of the 8th FPGAWorld Conference, 2011, p. 5.
[28] K. Squire, "Video games in education," Int. J. Intell. Games & Simulation, vol. 2, pp. 49-62, 2003.
[29] P. Jamieson, "Scotty in the Engine Room-A Game to Help Learn Digital System Design and More," in Meaningful Play 2010 Conference Proceedings, Michigan State University, USA, 2010.
[30] N. Kharma, L. Caro, and V. Venkatesh, "MagicBlocks: A game kit for exploring digital logic," in Proc. of the 2002 American Society for Engineering Education Annual Conference, 2002.
[31] K. Squire and H. Jenkins, "Harnessing the power of games in education," Insight, vol. 3, pp. 5-33, 2003.
[32] P. Sancho, J. Torrente, E. J. Marchiori, and B. Fernández-Manjón, "Enhancing moodle to support problem based learning. The Nucleo experience," in Global Engineering Education Conference (EDUCON), 2011 IEEE, 2011, pp. 1177-1182.
[33] M. Sanchez-Elez, "Introduction to VHDL Programming," http://eprints.ucm.es/27357/ 2011.
[34] P. P. Chu, FPGA prototyping by VHDL examples: Xilinx Spartan-3 version: Wiley. com, 2008.
[35] Xess Corp. "Using Xilinx WebPACK Software to Create FPGA Designs for the XSA Board" http://www.xess.com/static/media/tutorials/webpack-8_1-xsa.pdf 2005.

# A Simplified Sudoku 4X4 Game

**Simon H. Lin**
Department of Engineering and Computer Science,
Azusa Pacific University, Azusa, CA 91702, USA

**Abstract** - *Sudoku is a 9X9 board game. It has been so popular for its simplicity in rules, but it can be too complex for people to finish a game within an hour. This paper is to show a simplified Sudoku 4X4 game, which is so simple that people can finish a game within minutes. However, in order to win a game, the player should finish it within the time limit and without any mistake. In other words, the winning strategy of Sudoku 4X4 is different from that of Sudoku 9X9. This paper will illustrate the algorithm of this simplified Sudoku 4X4 game.*

**Keywords:** Sudoku, algorithm, GUI, Java.

## 1 Introduction to Sudoku Game

The original Sudoku is a 9X9 board game. The word "Sudoku" actually contains 3 Chinese characters: 數獨考. The first character means digit, the second unique, and the third examination. In other words, the game is an examination for player to put digits uniquely into the 9X9 board. This game allows only nine digits: 1 through 9 with 0 excluded. The 9X9 board has 9 rows, 9 columns, and 9 3X3 squares. The Sudoku 9X9 game has only three simple rules as follows:

1. Each row has all 9 digits with no repetition
2. Each column has all 9 digits with no repetition
3. Each 3X3 square has all 9 digits with no repetition

The rules are very simple, but the game can be very difficult since the board has 81 grids, and each grid can have any of the 9 possible digits. In other words, the game space is $9^{81}$. Please note both diagonals do not have to satisfy the unique rule. In fact, it is almost impossible for either diagonal to satisfy the unique rule.

Normally, a given game will have 30 grids pre-filled, and the player needs to fill the 51 grids with the correct digits. Sometimes, a game may have multiple solutions.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | 8 | | | | 6 | | |
| | 7 | | 6 | 3 | 4 | | 2 | |
| 2 | | | | | | | | 9 |
| 3 | | | 1 | | 7 | | | 2 |
| | 4 | | | | | 5 | 3 | |
| 6 | | | 3 | | 9 | | | 8 |
| 5 | | | | | | | | 7 |
| | 1 | | 9 | 8 | 2 | | 5 | |
| | | 3 | | | | 2 | 9 | |

The above board is a sample Sudoku game, and its solution can be as follows:

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 4 | 5 | **8** | 2 | 9 | 1 | **6** | 7 | 3 |
| 1 | **7** | 9 | **6** | **3** | **4** | 8 | **2** | 5 |
| **2** | 3 | 6 | 5 | 7 | 8 | 4 | 1 | **9** |
| **3** | 8 | 5 | **1** | 4 | **7** | 9 | 6 | **2** |
| 9 | **4** | 7 | 8 | 2 | 6 | **5** | **3** | 1 |
| **6** | 2 | 1 | **3** | 5 | **9** | 7 | 4 | **8** |
| **5** | 9 | 2 | 4 | 6 | 3 | 1 | 8 | **7** |
| 7 | **1** | 4 | **9** | **8** | **2** | 3 | **5** | 6 |
| 8 | 6 | **3** | 7 | 1 | 5 | **2** | **9** | 4 |

As you can see, a player may not finish this game within an hour. That's why we need to simplify the game.

## 2 A Simplified Sudoku 4X4 Game

The board for this simplified Sudoku 4X4 game is 4X4 instead of 9X9. The 4X4 board has 4 rows, 4 columns, and 4 2X2 squares. This game allows only 4 digits (or symbols) that can be pre-set before the game starts. Similar to Sudoku 9X9, the Sudoku 4X4 game also has only three simple rules as follows:

1. Each row has all 4 symbols with no repetition
2. Each column has all 4 symbols with no repetition
3. Each 2X2 square has all 4 symbols with no repetition

The rules are simple, and the game is much easier than Sudoku 9X9 since the board has only 16 grids, and each grid can have any of the 4 possible symbols. In other words, the game space is only $4^{16}$, which is much smaller than $9^{81}$.

Normally, a given game will have 6 grids pre-filled, and the player needs to fill the 10 open grids with the correct symbols.

The following is a sample game board of Sudoku 4X4:

| | | | |
|---|---|---|---|
| | 2 | | 4 |
| | | 1 | |
| | 3 | | |
| 2 | | 4 | |

The solution board may be as follows:

| | | | |
|---|---|---|---|
| 1 | **2** | 3 | **4** |
| 3 | 4 | **1** | 2 |
| 4 | **3** | 2 | 1 |
| **2** | 1 | **4** | 3 |

As you can see, a player may finish this game in a minute. The winning regulation for this Sudoku 4X4 must be more rigid to make this game fun and still challenging.

# 3 Winning Regulation for Sudoku 4X4

To win a Sudoku 4X4 game, the player must finish the game with no mistake on any grid within 40 seconds. In other words, an erase and redo of a grid is considered a mistake.

A different winning regulation is to use points. The following is a common way for points calculation. To put a symbol into an open grid is one move, and to put a different symbol into a grid (as a correction) is also considered as one move. If the player can finish the game in 10 moves, 10 points (the highest & the maximum) will be awarded. If the player finishes the game in K moves, (20-K) points will be awarded. If K < 20, the player would receive some points. If K > 20, the player would receive negative points as penalty. A top player is one who can earn 40 points for 4 games straight, or 50 points for 5 games straight.

# 4 Adding Difficulties to Sudoku 4X4

There are many ways to add difficulties to Sudoku 4X4 game.

Method 1 is to keep changing the 4 symbols from one game to another. For example, the 4 symbols of game 1 is {1, 2, 3, 4}, game 2 {A, B, C, D}, game 3 {5, 6, 7, 8}, game 4 {E, F, G, H}, game 5 {H, S, D, C}, and game 6 { ♥, ♣, ♦, ♠ }. The symbol change will add some difficulties to the game because it is now more difficult for players to memorize the solution patterns between or among symbols.

Method 2 is to consider time as a factor of penalty points. For example, each game allows only 30 seconds. For each additional 5 seconds being used, one point will be deducted.

Method 3 is to add the 4th rule: each diagonal has all 4 symbols with no repetition. This additional rule will definitely add some difficulties to the game.

# 5 Diagonal Rules for Sudoku 4X4

Both Sudoku 9X9 and Sudoku 4X4 have the same 3 rules to validate rows, columns, and squares. The Sudoku 4X4 game may have one more rule: each diagonal has all 4 symbols with no repetition.

| | | |
|---|---|---|
| **4** 1 2 3 | **4** 3 2 1 | **4** 3 2 1 |
| **3** 2 1 **4** | **2** 1 3 **4** | **1** 2 **4** 3 |
| **1** 4 3 2 | **1** 2 4 3 | **3** 4 1 2 |
| **2** 3 4 **1** | **3** 4 1 **2** | **2** 1 **3** 4 |

For example, the board above on the left satisfies the diagonal rule, but the other two do not. This diagonal rule does not apply to Sudoku 9X9 game.

# 6 Project Assignment for Data Structure Course

The following can be used as a project assignment for Data Structure course of Computer Science or Computer Information Systems.

Please write a Java program with GUI for the Sudoku 4X4 game. Your program must have main method so that .jar file can be created for your Java GUI program. You must submit .jar file as part of your project in addition to your source code and testing output.

This game allows only 4 symbols. The 4 symbols can be 1, 2, 3, 4; or A, B, C, D; or P, Q, R, S; or D, S, C, H (Heart, Spade, Diamond, Club); or ♥, ♣. ♦. ♠. Your program must keep changing the 4 symbols from game to game. The reason to keep changing the 4 symbols is to prevent players from memorizing the solution patterns.

| | |
|---|---|
| _ 2 _ 4 | 1 **2** 3 **4** |
| _ _ 1 _ | 3 4 **1** 2 |
| _ 3 _ _ | 4 **3** 2 1 |
| 2 _ 4 _ | **2** 1 **4** 3 |

4 Symbols: 1 2 3 4

The above Sudoku 4X4 board on the left (with 6 grids pre-filled by your program) is one example of the game. The above board on the right is a solution to the game.

The 4 symbols must be displayed to the player so that he/she can click one of symbol, and click where to put that symbol into the game board. The player is NOT allowed to change those **6** pre-filled grids, but he/she can keep changing or correcting any of those 10 open grids. A top player should do only 10 moves to fill those 10 open grids correctly in order to get 10 points (full maximum award).

There are 14 constraints for your program to check: 4 rows, 4 columns, 4 squares, and 2 diagonals. Each row/column/square/diagonal must have all 4 symbols with no repetition.

Bonus: You may add LEVEL to your program. LEVEL can be "Beginner" or "Advanced". If the player selects "Beginner" level, the 2 diagonal rules will not be enforced, otherwise (i.e., "Advanced" level) they will be enforced.

Your program will provide a Sudoku 4X4 game board with **6** grids being pre-filled. If the player can finish the game in 10 moves, 10 points (the highest & the max) will be awarded. If the player finishes the game in K moves, (20-K) points will be awarded. If K < 20, the player would receive some points.

If K > 20, the player would receive negative points as penalty. A top player is the one who can earn 40 points for 4 games straight, or 50 points for 5 games straight.

Please have your program dynamically generate a different valid game (with no apparent repetition) using different symbols from its previous game.

Your program must be fully tested by running at least 3 different games (using 3 different set of symbols). For each game, you must output the complete game sequence from the beginning to the end. Your program must be fully-documented with proper block comments and line comments.

# 7 References

[1] Anderson, Julie | Franceschi, Herve (2015) "*Java Illuminated – An Active learning Approach",* Fourth Edition, Jones & Bartlett Learning, www.jblearning.com, pp 1240.

[2] Prichard, Janet | Carrano, Frank (2013) "*Data Abstraction & Problem Solving with JAVA Walls & Mirrors",* Third Edition, Addison Wesley, pp 935.

# SESSION

# TEACHING METHODOLOGIES, CURRICULUM AND COURSE DEVELOPMENT + ADAPTIVE LEARNING SYSTEMS + ONLINE AND MASSIVE OPEN ONLINE COURSES

## Chair(s)

**TBA**

# A Strategy for Teaching "Real" Computer Science through Wearables

**Michael Eisenberg**[1], **Swamy Ananthanarayan**[1], **and Katie Siek**[2]
[1]Department of Computer Science, University of Colorado, Boulder CO USA
[2]Department of Computer Science, Indiana University, Bloomington, IN USA

**Abstract** - *The past two decades have seen a remarkable growth in the versatility, affordability, and aesthetic range of computationally-enhanced textiles and wearable devices. This burgeoning interest in wearables offers significant potential for computer science education: programming one's clothing and accessories might have obvious motivational appeal for students. On the other hand, there are* intellectual *obstacles to the project of founding computer science education on the use of wearable devices: the actual programming required for most projects is simply too restricted. What is needed, then, is a strategic framework for incorporating e-textiles and wearables into a more varied, challenging, and content-rich view of computer science education. This paper describes such a framework for* fungible programming *that envisions a hybrid of wearable and desktop/tangible computing as a basis for a "real" computer science curriculum. We illustrate this idea with examples from an ongoing educational project in which children employ programmable wearable devices to monitor their outdoor activity.*

**Keywords:** Fungible programming, e-textiles, programmable wearables, computer science education

## 1 Introduction: The Problem of Wearables in CS Education

Over the past two decades, a remarkable subculture of computer science has emerged, focusing on the integration of accessible, programmable computational elements into clothing and other textile-based artifacts. Hobbyists, crafters, and students make use of such devices as the LilyPad Arduino [2,3,5] and its offshoots (such as the Flora [7]) to incorporate dynamic behavior into hats, jackets, T-shirts, and a myriad other wearable artifacts [4,8, 10]. A perusal of websites such as YouTube or Instructables will serve to demonstrate the variety of creative projects incorporating these devices, and will likewise illustrate the enthusiasm with which young people in particular have taken to the idea of "wearable programming".

The advent of this subculture has been enabled by several concurrent technological developments: the availability of lightweight embedded computing devices (along with their associated programming platforms), the wide variety of sensors and actuators compatible with these devices, and (not to be overlooked) the arrival of materials such as conductive threads, tapes, and inks that enable users to connect together these various elements in creative craft projects. One might (for instance) create a shirt that lights up in various colors in response to sound, or a glove that signals the wearer in the presence of infrared light, or a knitted scarf that emits tones in response to a change in temperature; indeed, such projects are just the sorts of things that appear in numerous incarnations on hobbyist websites.

Projects of this sort hold powerful motivational appeal for computer science students–and, by implication, for computer science educators. Rather than working on abstract, "make-work" problems–writing an algorithm that sorts lists of numbers in ascending order, or searches for a string in a body of text–one might imagine introducing computer science and programming through a task such as making an interactive Halloween costume, sports uniform, ballet dress, or cosplay outfit. Programming one's own clothing taps into a cultural tradition of fun, creative display that has been associated with youth culture for decades. While a student might regard typical computer science assignments as "schoolwork", the challenge of making a programmable theatrical costume might be perceived as having a personal, idiosyncratic, irreplaceable purpose.

From the standpoint of the computer science educator, then, we should be vigorously exploring opportunities for incorporating e-textiles and wearables into computer science education, if for no other reason than to encourage students to (willingly) do their homework. Yet at the same time, there are significant obstacles to realizing this educational potential. Most pointedly, the difficulty is that there are profound *intellectual* gaps between the sort of programming issues raised by e-textile programming. A computer science educator wishes to introduce students to strategies for programming at the large, or at least medium scale–ideas such as modules, problem decomposition, object-oriented programming, debugging, and so forth. These are strategies that surface when students attempt programming projects of at least moderate size.

Few e-textile projects achieve this sort of size or complexity. It is fair to say that the vast majority of such projects have a relatively simple structure–something along the lines of "wait for a trigger sensor value, then turn on an actuator". The e-textile projects mentioned earlier all have this basic structure: one might wait for a temperature value above a certain threshold value, at which point the program sends a signal to turn on an LED light. From the CS educator's standpoint, a project of this sort might have value as an introduction to language syntax (e.g., conditionals), but little else.

What is needed, then, is a strategy for exploiting the motivational and aesthetic potential of e-textile and wearable computing while, at the same time, leading toward projects of greater complexity and challenge. This paper describes such a strategy, which we call *fungible programming*, and illustrates it with (still relatively early) examples derived from an ongoing project centering on children's fitness and health. The basic idea behind fungible programming is to create "hybrid projects" in which wearables are treated not, or not only, as reactive display items, but as mobile sensors and actuators loosely linked to larger projects in desktop or Web programming. The term "fungible" here is used to highlight the idea that the programs designed for wearable devices create and employ data that may be freely exchanged, for a variety of purposes, with other programs and devices. Rather than viewing wearable devices as the sites of (tiny) standalone programs, as is true of most projects, we can view them as the mobile "outer skin" of complex software systems, as suggested by the diagram in Figure 1.

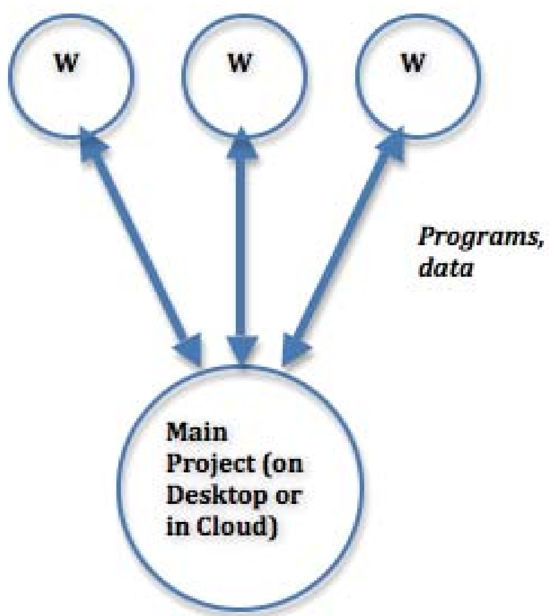



Figure 1: A basic framework for "fungible programming" of wearable devices. In typical projects with wearables, the "main" desktop computer endows the wearable devices ("W") with programs that work autonomously. In fungible computing, the wearable devices interchange data with the desktop computer or "main" project. The result is that the wearables are "extensions" into the world of the larger project.

The remainder of this paper is organized as follows: in the second section, we introduce several working examples that suggest (though they hardly exhaust) the ideas behind fungible programming. We also explore the application of these ideas to educational programming by describing recent work in combining wearables and a group project conducted in a local after-school computer clubhouse. The third section goes beyond these initial examples by outlining (perhaps mildly futuristic) scenarios for a fuller computer science curriculum based on e-textiles and fungible programming. Here, we address conceptual issues that effectively extend the traditional portrait of the by now "classic" Von Neumann architecture for computers. In the final section, we discuss some of the opportunities for evolving computer science education in a more craft-based direction, and some of the obstacles that must be overcome in this evolution.

## 2 Fungible Programming: Some Initial Prototypes

In this section we concretize the framework described in the previous section by showing several working prototypes of the notion of fungible programming. For our examples, the wearable component consists of three modules: a microcontroller base, a sensor module for ultraviolet light, and a button that the user may employ for manual signals.[1] The base is a custom printed circuit board of our own design, with our own surface mount components, designed to be lightweight and to lay flat against various (presumably textile) surfaces; the microcontroller may be programmed through the (by now fairly "standard") Arduino programming environment, which is both freely available and well-supported by a large community of users.

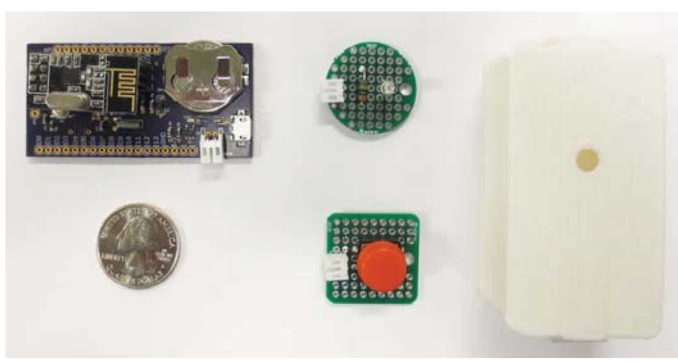


Figure 2: Clockwise from upper left: wearable base with coin cell battery and RF wireless module; UV sensor; 3D printed case to house the device; wearable button.

[1] A description of an earlier version of this wearable device can be found in [1].

The basic idea behind this particular wearable design is grounded in our own work in allowing youngsters to track their outdoor activity. The UV sensor is, in effect, a means of keeping track of the time that the wearer spends outdoors. The microcontroller base records the passage of time by using a watch crystal and the microcontroller's real time counter; when connected to the UV sensor it periodically timestamps events and stores the timing data in memory. In this way, the device may be regarded as a "monitor for time spent in sunlight." Because of this particular device's low weight and light power requirements, it may be run on a single coin cell battery for weeks. Figure 2 shows the various components of the system, including a plastic case that we designed and printed on a 3D printer.

It should be noted that the design choices for our own wearable system are motivated by a specific, health-related project; but the larger discussion of fungible programming in this paper does not depend on any one design for wearables. Rather, as we will discuss later, the idea of fungible programming can be tested and explored with an endless variety of specific choices and designs for wearable components. Our own initial experiments, however, are based on the wearable system shown in Figure 2.

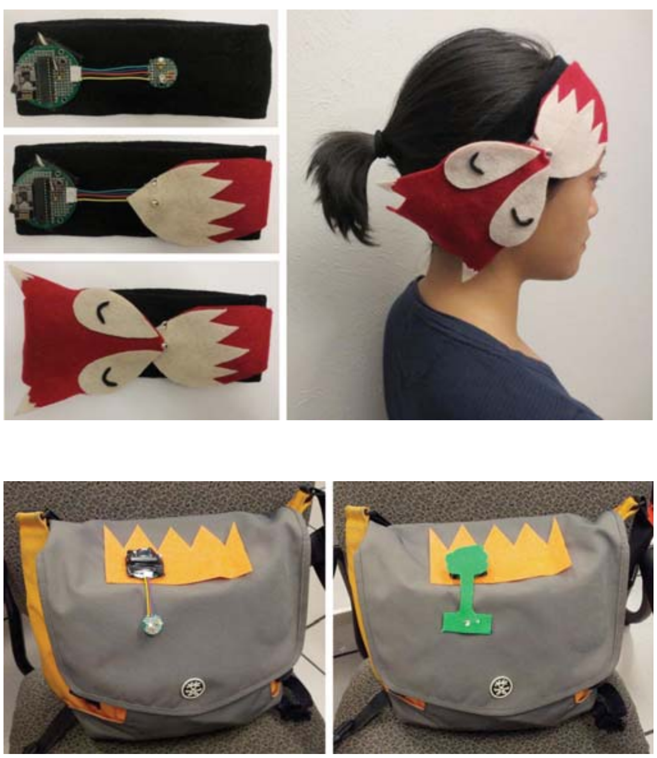


Figure 3. Two working examples of the "wearable" elements of a hybrid project approach. Here, a somewhat earlier version of our wearable device (see [1]) has been incorporated within a headband (top) and a messenger bag (bottom); both projects make use of felt and Velcro as basic craft materials. These items may be used to gather data on time spent outdoors which may then be communicated to the larger, stationary devices of Figure 4.

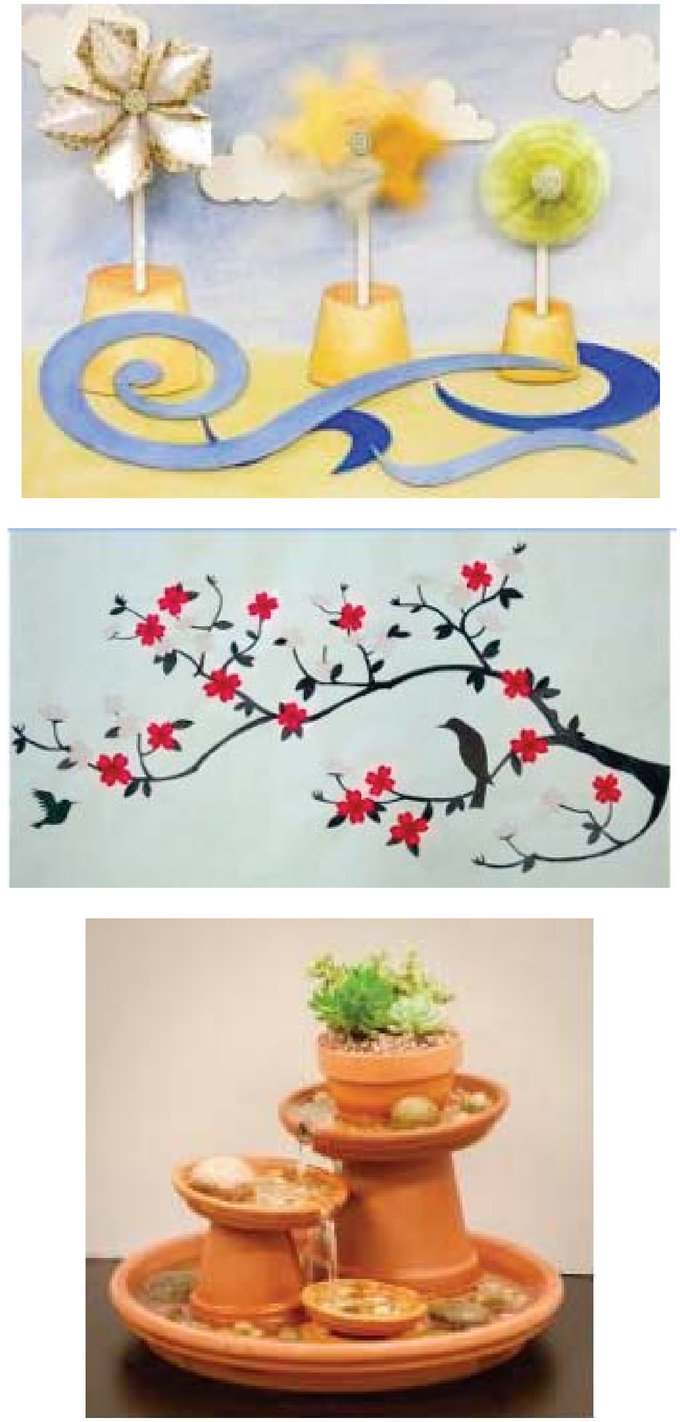


Figure 4: Several working examples of the "non-wearable", stationary elements of a hybrid project approach. At top, a windmill kinetic artwork in which the rate of the turning mills can be controlled from data provided by the wearable. At center, a cherry blossom painting in which the number of blossoms that light up with LEDs is correlated with data from the wearable device. At bottom, a terra cotta fountain whose flow rate is controlled by data from the wearable.

The wearable device shown in Figure 2 may now be incorporated into clothing or accessories of the sort shown in Figure 3; a headband or messenger bag could now be used to record the amount of time spent outdoors by the user. Indeed, the sorts of artifacts in Figure 3 represent the typical sum total of most published wearable projects; in a standard scenario, the wearable device might also be connected to an LED light that illuminates when (e.g.) the UV level, or time spent outside, exceeds a given threshold.

The fungible programming approach now uses artifacts such as those shown in Figure 3 as the data-gathering "input devices" for more elaborate projects of the sort shown in Figure 4. For these projects, the basic idea is that the amount of time spent outdoors can be employed as a sort of "virtual currency" for powering computationally-enhanced craft items created at home. Once the user returns home, she sends the data from the wearable device via a wireless radio frequency signal (RF) to a microcontroller (or multiple controllers) embedded in the craft object itself; the home object may then respond in specific ways to the data conveyed by the wearable device. In the examples shown in Figure 4, a windmill portrait can turn faster in response to more time spent outdoors; a cherry blossom painting will respond with red LED lights (and a chirping bird) to data from the wearable; and a terra cotta fountain will increase its flow rate in response to more time spent outdoors.

A detailed discussion of the implementation of the Figure 4 objects is tangential to the main point of these examples (a thorough discussion of the cherry blossom painting may be found in [1]). For the purposes of this discussion, the larger issue is simply that these projects illustrate a hybrid of typical wearable projects (illustrated by Figure 3) and more complex computational projects for the home (illustrated by Figure 4). There is an obvious combinatorial advantage to this sort of approach: for example, either of the two wearables in Figure 3 could be combined with any of the craft projects in Figure 4, so that even in these very initial illustrations we have effectively shown six potential projects. More broadly, while our own laboratory's interests focus on craft projects (of the form shown in Figure 4) as the representative of "complex, stationary" systems, there is no reason not to expand the scope of this vision to (e.g.) a project run on a desktop computer, on a Web server, or (more generally) as a cloud-based system. In section 3, we will provide potential examples of this broader vision of fungible programming.

## 2.1 Student-Built Prototypes

A natural objection (or at least, source of skepticism) to the vision sketched above is that the fungible programming approach might prove too complex, or too demanding of attention span, to justify its use as the basis of a computer science curriculum. While this is an empirical question to be explored in practice, our own preliminary experiences with middle-school-age students gives us optimism that in fact the notion of fungible programming could prove feasible as a foundation for computer science education, even for relatively young learners.

During the past year, we worked with nine students between the ages of 11 and 14 over a period of several months at a local after-school enrichment program (focused on arts and technology). While a full description of our study is beyond the scope of this paper (and is the focus of another paper, in preparation, largely concerning the results of the students' work), the purpose of mentioning this work here is merely to illustrate the plausibility of fungible programming for younger students. In our after-school group, students created their own wearable artifacts incorporating the (newer) system shown in Figure 2 earlier. Figure 5 shows two representative examples of this phase of the students' work: a pencil pouch with a flower design, and a sports-themed book cover.

Figure 5: Student-designed accessories using the wearable device shown in Figure 2. At top, a pencil pouch with a felt flower design (the microcontroller is hidden behind a leaf). At bottom, a book cover in which the wireless module extends through the felt of which the cover is made.

Again, it is worth noting that many "traditional" studies (if that word can be applied to such an embryonic technology!) of wearable computing for children would have effectively ended with the artifacts shown in Figure 5. In this

Figure 6: Electronic mural created by middle school children for combination with the wearable modules in Figure 5. Each individual segment of the mural reacts to data from its particular owner's wearable device. (and sent via wireless RF signal). To take several examples: on the left of the mural, the galaxy spins using an embedded motor, and the boy seated on the beach plays a tune using a speaker. On the right, the mask vibrates and the octopus figure changes color.

project, however, the wearable elements were intended to gather data for use with a larger, in-class project–the mural created by the children and shown in Figure 6. Each portion of the mural was created with a specific child's wearable device in mind; for example, the boy sitting on the beach at the bottom left of the mural plays a musical tune in response to the data sent from its creator's wearable device.

Again, there are many more observations to be made about the children's work and its relation to their conceptions of fitness and health; but for the purposes of this discussion the crucial point is that a "hybrid" approach to children's technology, in which (relatively) complex classroom projects are linked to wearables, is communicable to students. It would be precipitate to claim on this basis that a fuller computer science education could be founded on this strategy; the following section presents a more thorough (if speculative) argument to that effect.

# 3 Designing a CS Curriculum Based on Fungible Programming

The previous section presented a variety of working projects that suggest the plausibility of a "true" computer science curriculum based on fungible programming. This section, more frankly speculative, is intended to outline the sorts of activities that such a curriculum might entail.

The essence of the "fungibility" notion is that a variety of devices, independently programmed, can communicate results to one another in a natural, interchangeable way for a variety of projects beyond the scope of any one device. There is, it should be noted, nothing conceptually startling about this idea: one could view JPEG and STL formats, for instance, as multi-platform data files for use across many projects in just this sense. The purpose of emphasizing the notion here is that the burgeoning world of hobbyist devices–prominently including wearable devices, embedded microcontrollers, RF tags and readers, and the like–have not been incorporated into an educational framework for computer science in the way that they could. For the most part it is fair to say that the standard model for working with these devices is that they are first programmed from a desktop machine, and then placed into situations (like a "light-up jacket") in which they do their work autonomously. Wonderful as many of these examples are, they constrain the possibilities for education, since (as already noted) the programs associated with the vast majority of projects are limited in complexity.

A true CS curriculum for devices of this sort (and here we focus on wearables) should instead regard the hobbyist items as small-scale computers in their own right, capable of communicating back and forth with one another and with desktop devices. In a sense, then, rather than conceive of wearables as creating "dynamic clothing", we can view them as extending the capacity and range of the classical input and output "boxes" in the standard Von Neumann architecture. Wearables can be viewed as a type of "intelligent skin" for more complex programs created in classroom settings.

A few suggestions for how this idea might be implemented follow; and the reader is encouraged to brainstorm similar possibilities for him- or herself.

• One possibility might be to employ "tuned" wearables as the basis of a computationally-enriched scavenger hunt (not unlike the basic structure of "geocaching" activities). For instance, a sound sensor might first be programmed to seek out different frequencies (perhaps out of the range of normal human hearing), and the students' jobs would be to gather instances of these frequencies from their community. Once gathered, the job of the students would be to design a display program that (e.g.) would show the locations or distributions of the gathered frequencies on a desktop screen. The purpose of such a project would be to combine the strengths of wearables with a more challenging problem in data display. Similar ideas might be implemented with (e.g.) light or chemical sensors rather than sound sensors as a foundation. For instance, students might "gather" colors from outside that could then be added to a graphical palette.

• A machine learning project could be founded on programming simple three-layer neural networks into wearable devices so that students could now go out into the world and train the devices to respond to particular words, colors, or voices. For instance, a wearable might be equipped with a student accessible button (in fact, our system in Figure 2 employs just such a device); when the wearable correctly recognizes a voice (by, say, lighting a green LED) the student reinforces the network, and when it fails to do so the student induces a revision of the network weights. Over time, then, the student's wearable should become a completely individual device that recognizes only (say) the voices of selected people, or selected code words. Here, an introduction to machine learning can be naturally combined with the playful individuality of a wearable device.

• Students might design wearables that can communicate with each other over short distances, forming themselves into larger multi-person "display screens". For instance, a set of students positioned in a line might program their wearables to behave like individual cells in a one-dimensional cellular automaton. By standing nearby one another, the line of students could create interesting graphical patterns. Such a project could be accommodated to relatively small or large groups of students, and perhaps to more complex patterns of communication (e.g., two-dimensional arrays). (For an early, pioneering effort along these lines, see Colella's [6] work in "participatory simulations.")

• Much as the students in our own clubhouse designed a "communal mural," one could imagine a classroom of computer science students creating a highly ornate or complex "communal wearable" to which all students contribute over time. Wearing this garment could be thought of as a sort of "prize" (think, for an analogy, of the coveted "yellow jersey" worn by the leader of the Tour de France bicycle race). The CS teacher could thus design an activity in which (e.g.) for every practice problem completed successfully, a personal LED is added to the classroom garment; and from time to time, individual students (as a sort of playful reward) are given a chance to wear the garment. This, again, might be a strategy for integrating the joy of personal display with the (arguably more abstract or mundane) tasks of the day-to-day computer science classroom.

Activities of this sort all seem to us implementable, and only experience will determine their educational value. Nonetheless, it should be noted that the design of projects of this sort are compatible with many (if not all) the standard elements of an introductory CS curriculum. Activities such as these could be blended with an introduction to various syntactic forms (conditionals, iteration); or with programming of graphical or musical data; or with ideas of graph search, machine learning, cryptography, pattern matching, emergence, and so forth. Perhaps most interesting is that the aforementioned notion of wearables as an "intelligent, computationally-enriched skin" or sensory membrane surrounding a core project could bring students to challenging concepts that do not always make their way into introductory CS courses–notions such as parallel or multiprocessor programming (the "cellular automaton" example mentioned above might involve simple concepts along these lines, as individual wearables would have to be synchronized to enable the project to work).

## 4 Related Work, and an Affective Rationale for Wearables in the CS Curriculum

The excitement of wearable computing has already become apparent in hobbyist culture, and seems to hold particular appeal for the young. Numerous recent books such as [4], [5], [8], and [10] speak to the creative ferment around these devices. Moreover, from the standpoint of demographics, accessible wearables seem to attract new populations to computer science, as noted by Buechley and Mako Hill [3] in an early study of the use of the LilyPad Arduino. In particular, these devices seem to hold strong (though not exclusive) appeal to young women and girls, who are traditionally underrepresented in computer science courses. While some computer science educators have sought motivational appeal in topics such as game design [11], robotics [9], and computer graphics [12], adding wearables to this constellation of "core motivating activities" could attract still different groups of students to a lifelong interest in computing.

More generally, incorporating wearables into computer science education suggests something of a different "lens" through which to view many of the debates that historically characterize this field. Rather than argue over (say) choice of language, or textbook, or the core notions of "computational thinking" (this last seems to be an endless source of argument in our own community), we could begin rather with a *narrative* or *biographical* view of the student's purposes in going into computer science. That is: who in fact does the student wish to be, or become, through learning computing?

Typically, educators answer this question in the least interesting way possible: by assuring students that a (presumably corporate) job awaits them at the end of all this effort. Not surprisingly, the question of acquiring a job might motivate some students (especially older ones) but it provides at most a thin, potentially grim-sounding future for many others. Merely telling students that they can spend a lifetime earning a salary (or worse, "competing with China") is, in effect, attempting to graft the motivations of *employers* or executives onto the lives of creative students.

In general, youngsters are attracted to computer science (when they in fact are attracted) by a vision of who they can be–of what *type* of person, what type of life, is associated with this study. A computer science education that includes, or focuses on, wearables not only has *intellectual* benefits (introducing embedded computing, distributed processing, and the like), but it has *affective resonance* for many students. A youngster can wear and display his or her own interests among their peers, in theatrical productions, in slam poetry contests, in sporting competitions, at dances. Computing means something to many students in these contexts. A CS curriculum that incorporates these devices–and that employs the strategy of fungible programming to bring wearables into the world of advanced computer science–stands a fair chance to make a difference in many students' lives, and (we can hope) in the larger practice of CS education.

## 5 Acknowledgments

The work described in this paper was partially supported by the National Science Foundation under award IIS-1231645.

## 6 References

[1] S. Ananthanarayan, N. Lapinski, K. Siek, and M. Eisenberg. [2014] Towards the crafting of personal health technologies. In *Proceedings of the 2014 Conference on Designing Interactive Systems (DIS 14)*, pp. 587-596.

[2] Buechley, L., and Eisenberg, M. [2008] The lilypad arduino: Toward wearable engineering for everyone. *Pervasive Computing*, IEEE 7:2, 12–15.

[3] Buechley, L. and Mako Hill, B. [2010] LilyPad in the Wild: How Hardware’s Long Tail is Supporting New Engineering and Design Communities. In *Proceedings of Designing Interactive Systems (DIS) 2010*, pp. 199-207.

[4] Buechley, L.; Peppler, K.; Eisenberg, M.; and Kafai, Y. [2013] *Textile Messages*. New York: Peter Lang.

[5] Buechley, L. and Qiu, K. [2013] *Sew Electric*. Cambridge, MA: HLT Press.

[6] Colella, V. (2000) Participatory Simulations: Building Collaborative Understanding Through Immersive Dynamic Modeling. *Journal of the Learning Sciences*, 9:4, pp. 471-500.

[7] Flora. wearable electronic platform. At: www.adafruit.com

[8] Hartman, K. [2014] *Make: Wearable Electronics*. Sebastopol, CA: Maker Media.

[9] Kumar, D. [2011] *Learning Computing with Robots*. Text available at: http://wiki.roboteducation.org/Learning_Computing_With_Robots

[10] Peppler, K. and Grisalfi, M. [2014] *Soft Circuits: Crafting E-Fashion with DIY Electronics.* Cambridge, MA: MIT Press.

[11] Repenning, A.; Webb, D.; and Ioannidou, A. [2010] Scalable Game Design and the Development of a Checklist for Getting Computational Thinking into Public Schools. In *SIGCSE 2010 Proceedings*, pp. 265-269.

[12] Tanimoto, S. [2012] *An Interdisciplinary Introduction to Image Processing.* Cambridge, MA: MIT Press.

# A Study of Kinesthetic Learning Activities Effectiveness in Teaching Computer Algorithms Within an Academic Term

Matthew Lai, Danny Luong, and G. Young
Computer Science Department, California State Polytechnic University, Pomona, CA 91768

## Abstract

*Kinesthetic learning is a teaching method that involves students' physical interaction among each other and the environment. The typical method of teaching, a classroom with an instructor talking and students listening and taking notes, has been the norm for centuries. This research attempts to show that the kinesthetic learning activities(KLA) approach can be a viable alternative. In this study, the performance of students from an undergraduate level computer science course, Parallel Processing ,is considered. In the Spring 2014 quarter, the class was divided into two groups. Each group was alternatively taught using KLA and traditional methods, allowing us to gauge the effectiveness of the KLA approach in one quarter. By performing this test over the course of a single quarter, we hope to more definitively show the effectiveness of a KLA approach to teaching students. The students' gained knowledge was measured through pre / post tests. We hypothesized that the KLA approach would be as efficient as the traditional lectures if not more efficient. The data that was collected from these tests favor our hypothesis.*

**Keywords:** *kinesthetic, learning styles, teaching, computer science, algorithms.*

## 1. Introduction

In the present day, technology is everywhere and in the hands of everyone. Although the benefits of technology in the learning process are immeasurable, technology itself has many drawbacks. One of these drawbacks is the increase in distraction among students. In most classroom settings, students just sit and listen to an instructor. It is easy for them to lose focus by checking social media or chatting with their friends. In addition, several studies suggest that students' attention during a lecture may last only up to fifteen minutes [9]. Therefore, it is challenging for some students to maintain concentrationin a standard lecture.

People have different learning styles. According to Neil Fleming's model, there are four types of learners: The first group is visual learners who prefer learning through symbolic representations, using graphs and charts to obtain knowledge. Aural learners are those who perceive more knowledge when they are listening to a lecturer, and they have no problems learning in traditional lecture-based environments. There are also learners who prefer learning through text, either reading a book or writing notes. The last type is the kinesthetic learners who gain more knowledge through physical simulations for the concepts they are learning [4].

Active learning or kinesthetic learning approach is a solution for those who have trouble paying attention for a long period of time because it requires students to be physically active during the lecture. Since computer science topics are theoretical and intangible, grasping the concepts may be challenging for the students. Therefore, kinesthetic learning activities can be helpful and efficient.

### 1.1. Active learning characteristics

Active learners tend to learn better through physical activity. Hence, they prefer performing arts and athletics more than studying theoretical science. They get distracted easily in a traditional lecture environment. They perfectly apply to the Chinese proverb which says "*I hear and I forget. I see and I remember. I do and I understand.*" These learners gain more knowledge when they use their hands and bodies. The KLA teaching method provides a suitable learning atmosphere because it forces learners to pay full attention in their movements. For example, a role playing KLA allows learners to imagine a situation where a problem arises, and they act like the solving agent. Another example is playing a game in the classroom like playing the tower of Hanoi puzzle. Kinesthetic learning gives an exciting learning experience for all learners, especially active learners.

### 1.2. Active learning techniques

Faust and Paulson [3] have categorized active learning techniques into six types: First, exercises for individual students, which include quizzes that require the students to be in the classroom and pay attention to the lecture. Second, questions and answers, which is a method that allows the students to ask questions anonymously using the fish bowl technique, and the rest of the students answer these questions. This method empowers the students' role in their own learning process. Third, immediate feedback, which is exactly as its name suggests; it gives the professor an opportunity to measure the class's understanding as whole, and it is a helpful method that can be used in large classes. Fourth, critical thinking motivators, is a method that aims to get the students to think about the concept before they actually learn it using puzzles. The tower of Hanoi puzzle is a good example of this technique. The concept can be explained to students before they attempt to solve the puzzle. The learning process actually happens when they solve the puzzle with the algorithm that they have been provided. The fifth active learning type is share/pair. This method forces students to work together as pairs by exchanging their thoughts and opinions and explaining unclear concepts. An example of this learning type is pair programming which always produces high quality software [10]. The last technique is the cooperative-learning strategy, which requires students to be divided into groups. Each group is responsible of solving a given problem and explaining it to the rest of the class. This method enhances students' communication and social skills.

### 1.3. Benefits

The typical classroom setting, where students are just passively sitting and listening, leads students to enter their "relaxed zone," where it is much harder for them to maintain full attention. On the other hand, a kinesthetic activity fills the room with energy and excitement, and it makes students see things from a new angle. Moreover, a KLA also helps students to develop interpersonal skills, since it requires them to communicate with each other. It helps timid students to interact with their classmates in an observed environment. As reported by the National Training Laboratories' pyramid of learning, students gain only 5% of the information given to them in the form of lectures while learn-by-doing retains 75% of the knowledge. Furthermore, students retain 90% of what they have learned when they teach each other [7].

### 1.4. Obstacles

Despite the numerous advantages of active learning, some instructors are reluctant to adapt this teaching method for several reasons. For example, how an instructor decides to manage their time has a major influence of thewhole learning process and its outcome. An instructor must account for the preparation and execution time, as well as students' responses to such new learning strategy. Some instructors think that they do not have enough time to cover all the topics they have assigned for a certain quarter, and active learning can reduce the amount of available time. As a result, an instructor may conclude that lecturing is more convenient for delivering the information. However, a scenario where an instructor explains and students listen does not guarantee that students will be able to absorb the knowledge. Students may leave their classroom with some bits and pieces of a lecture in their notebooks and with nothing in their heads. Also, there is a big chance that a number of students may stop the instructor to ask questions, and the instructor might end up not covering every topic that needs to be covered. In addition, in a traditional lecture, the teacher has more control of the class where in an active learning session; students are more involved with how the course is run. With this in mind, an instructor may be ambivalent to use KLA as new problems with classroom management may arise.

## 2. Past Research

In the past decade, kinesthetic learning activities were common in preschools and elementary schools. In 2004, Tammy Nguyen did a study named "Do kinesthetic strategies influence students' achievement?" She studied the effectiveness of teaching mathematics kinesthetically for first-graders. She taught them addition and subtraction using both a traditional method and a kinesthetic method using hand signals, and she tested them. The results showed that the kinesthetic test scores were higher than the non-kinesthetic test scores. Also, she noticed that the students were excited about participating in the KLA's. Though significant for its findings on the benefits of kinesthetic learning at the preschool and elementary level, this study by itself cannot prove that kinesthetic learning is more effective than traditional lecturing in general [6].

In 2009, Katherine Gunion taught middle school students the concept of recursion in a KLA fashion. She held a 7-weeks-after-school program for the students and used 6 different KLAs. She wanted to answer three questions:

1- Can students identify recursion?

2- Can the students understand and apply recursion?

3- What is the effect of understanding recursion on their attitudes towards computer science?

Her answers were as follows: most of the students were able to identify recursion, but few of them were able to apply it on solving different problems instead of solving them sequentially. For the last question, the different data collecting methods she used showed that the students enjoyed the activities, and half of them came back for the next offering of the program, but this still does not prove our goal [5].

Similarly, for college students, there were several publications about kinesthetic learning activities in the computer science field. "Human cons cell jeopardy" is such a KLA developed by Begel, Garcia, and Wolfman to introduce any programming course [2]. In 2007, Sivilotti and Pike developed a set of KLAs for both undergraduate and graduate students to teach them the concepts of distributed systems [8]. In general, there are only a few KLAs that have been designed for computer science education at the university level.

## 3. Research Goal and Methodology

In this study, we applied kinesthetic learning strategies on students enrolled in a Parallel Processing course in the Computer Science department at California State Polytechnic University, Pomona (Cal Poly Pomona) to test the efficiency of the KLAs compared to traditional lectures. Our hypothesis says that kinesthetic learning will be as effective as traditional lecturing, if not more so. It is important to mention that this research has been revised and approved by the Cal Poly Pomona Institutional Review Board (IRB): protocol #14-0004, and it has met the federal and state regulations and Cal Poly Pomona policies regarding the safety of the human subjects who participated in this research.

In Spring 2014, the students who attended the CS370 Parallel Processing class participated in this study. Similar to our previous study from the Winter and Spring quarters of 2014 [1], we presented the groups of students to both a traditional lecture on the selected topics as well as a KLA lecture. However, for this path of research, our study group consisted of only a single class from a single quarter. The chosen topics for this study were parallel pipeline algorithms for addition, prime number generation, insertion sort, and systems of linear equations and parallel algorithms for merge sort, bubble sort, radix sort, and shear sort. Prior to the lectures, the Parallel Processing class was given multiple pretests to gauge their background knowledge about these topics. The class was then divided into two groups and each group was alternatively taught using traditional and KLA methods over two sessions. The students were given a posttest after each lecture to determine their understanding of the topics after learning from either the traditional lessons or the KLA activities. To this end, we developed kinesthetic learning activities for the listed topics to study the benefits of using kinesthetic approach in Computer Science.

## 3.1. Parallel Pipeline Algorithms

### 3.1.1. Addition

For the addition pipeline KLA, we performed the calculations such that each participating student simulates a processor in a linear series of processors. The first student receives an instance of addition and performs an operation. That processor passes on the instance to the next processor in the pipeline while simultaneously receiving another instance of addition. Along the pipeline, the next processors performs an operation before passing on their results to the next processor. This demonstrates one of the usages of pipeline processors in processing multiple instances of the same operation, in this case multiple separate instances of addition.

### 3.1.2. Prime Number Generation

For our design of a KLA for prime number generation in a pipeline, we again have the students each act as a processor in a linear series of processors. Numbers of increasing value, beginning with 2, are passed one by one to the first processor in the series. As each processor receives a number, they do the one of the following:

1. If the student does not possess a number prior to receiving the new number, they retain the new number and perform no further actions for the round.
2. If the student does possess a number and the number that they receive is not a multiple of the number they possess, then the student passes the number that they have received to the next student at the beginning of the next round.
3. If the student does possess a number and the number that they receive is a multiple of the number that they possess, then they immediately drop the number that they received.

This process can continue until the instructor decides to end the KLA or the final processor in the pipeline obtains a number.

### 3.1.3. Insertion Sort

The insertion sort KLA that we implemented is a simple exercise. Once again the students are arranged such that they are each a processor in a pipeline series of processors. A number is fed to the first processor of the pipeline. If a processor receives a number and he does not possess a number then the processor retains the number and performs no further actions. If the processor receives a number and he does already possess a number, then the processor compares the two numbers.In the next round, it passes the smaller number to the next processor in the pipeline. At the end of the KLA the students should possess a series of numbers sorted in descending order starting from the head of the pipeline.

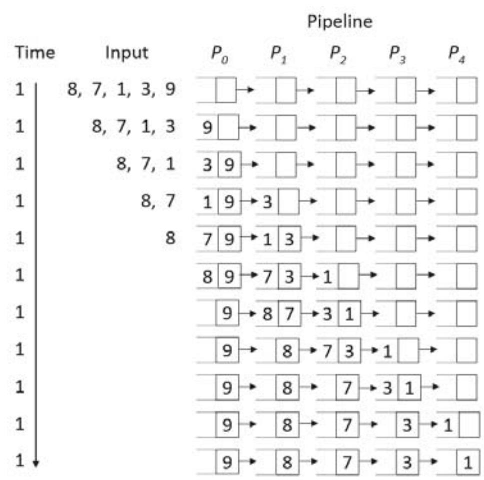



*Figure 3.1 Image demonstrating pipeline insertion sort.*

### 3.1.4. Systems of Linear Equations

A KLA for teaching how a system of linear equations can be solved using a pipeline of processors can be run as follows. Each student represents one stage of the system of linear equations pipeline algorithm and each stage is responsible for one equation. When a stage receives a variable that was previously unknown, it copies down the number and passes the number to the next stage in the pipeline. It then uses this variable to attempt to calculate its own unknown variable. If the processor is able to solve for the unknown variable, it passes it on to the next processor in the pipeline. This shows how a pipeline can be used to pass information further along the pipeline, even before the processor has completed all of its internal operations.

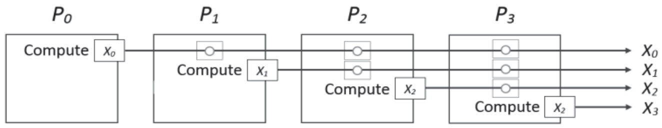



*Figure 3.2 Image demonstrating Pipeline System of Linear Equations*

## 3.2. Parallel Sorting Algorithms

### 3.2.1. Merge Sort

Because of the nature of merge sort, a KLA for running merge sort in parallel is fairly easy to design but not necessarily easy to implement with the students. In our design for a KLA of parallel merge sort, we have each student represent a single processor. We begin the KLA having already divided the array into the smallest elements for the algorithm.Each number of the array is represented by a separate piece of paper. During each round, each student only performs one comparison, during which it compares the numbers as required by the merge sort algorithm and swap them as necessary. Each subarray of numbers is then passed to the next processor in thetree and the merges are performed in parallel. At the end of the process, one of the students performs the final merging of the two largest subarraysat which point, there is a final sorted array.

### 3.2.2. Bubble Sort

Parallel Bubble sort, also known as Odd-Even Transposition Sort, is a relatively simple sorting algorithm designed to work on parallel processors. It is a comparison sort similar to a bubble sort that compares, alternately, the odd-even and even-odd adjacent pairs and swaps them if they are in the wrong order. The KLA we designed for this course utilizes each student as an individual processor. Numbers on adhesive paper are stuck to the wall and the students alternately compare their assigned even and odd pair then the odd and even pair and switches their placement as necessary. When no swaps are made in a given round, the sort is finished.The resulting array of numbers should be in sorted order.

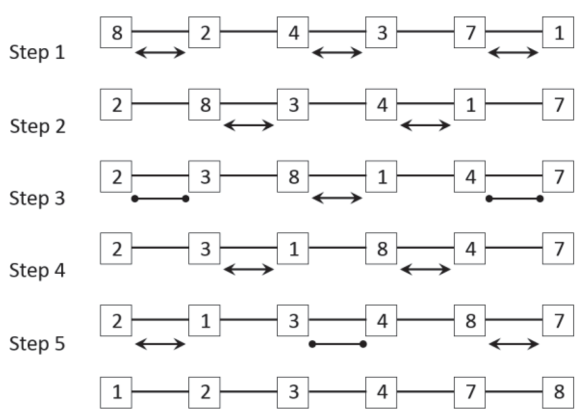



*Figure 3.3 Image demonstrating Odd-Even Transposition Sort*

### 3.2.3. Radix Sort

Radix sort is a non-comparative integer sorting algorithm that sorts each integer by grouping them according to individual digits of each integer. In the KLA we designed to demonstrate parallel radix sort, we utilized plastic cups as the "buckets" into which each element is placed. We used ping pong balls marked with integers to represent the integers for easier handling. For ease of sorting, we used integers of base four and maximum integer length of three. Each student acts as a processor and are arranged such that the first order of students, in our case only one student, processed each integer and passed the integer to the appropriate second order of students according to their most significant digit. Each succeeding order of students processed their given integers according to succeedingly less significant digits and passed them onto the next order of students until the least significant digit was processed and placed into the appropriate plastic cup. If the ping pong balls were removed from the cups in order, then the integers were shown to be sorted.

### 3.2.4. Shear Sort

Shear sort is an algorithm designed for parallel processing. Shear sort sorts a two-dimensional array into a snake like order within the array. It alternatively sorts the array row-wise, where the odd rows are sorted in ascending order and the even rows are sorted in descending order, and column-wise where all columns are sorted in ascending order downwards. For our KLA, we have the students stand together as an *n* by *n* two-dimensional array. Each student in the array holds a paper with an integer printed on it. There are then *n* student processors who alternates between sorting the array row-wise and column-wise. When there are no changes made in a round, the array is sorted in a snake like fashion.

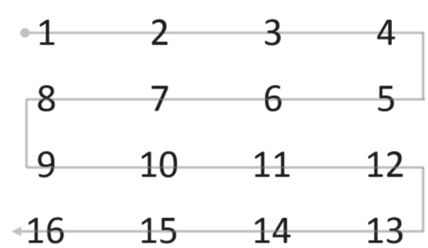



*Figure 3.4 Image demonstrating array sorted in snake-like order*

# 4. One Quarter Methodology

In what we believe to be a unique study, we have performed this KLA research using a single class of students to attempt to better gauge the effectiveness of KLA teaching methods. We have not found this particular line of research in any of the existing literature on KLA studies.We present our method of evaluating KLA effectiveness utilizing a single body of students, conducted over the course of a single quarter. To do this we had to solve a small number of problems.

### 4.1. Constraints and Requirements

Due to the nature of this research, there are certain constraints and requirements that must be understood and met to perform this study.The most important constraint is the requirement of two separate lecture areas. While the students of one group are learning some material with a traditional lecture, the other group is learning the material under the KLA method. We don't want to cross contaminate how the students learn the material. It is then essential that we have access to two separate rooms from which we could lecture the two groups of students separately. In our study, we had access to a room with an adjoining lab area with standing space, perfect for conducting KLA activities. The main classroom was a larger space with writing space where the students could be taught using traditional methods. It is also imperative to keep in mind what equipment is available in each room. The equipment, such as whiteboards, computers, and projectors are essential tools with which students can be taught. In our situation, our large main classroom is a widely used lecture room equipped with all the necessary lecture tools, making it easy to perform traditional lectures. However, the smaller adjacent space, in which we planned to lecture using the KLA method, is a lab area designed for testing hardware and software projects and did not contain a whiteboard or a projector. This classroom's deficiency in teaching ability was mitigated by using a loaned projector cart with which we could project our lectures onto the wall.

The next most important resource required for this study was the instructors who would teach the material. For this study, we needed two comparable instructors who are capable of performing both a traditional lecture and a KLA lecture. Because of these requirements and the time of day at which the classes were held, it was decided that two graduate students would perform the lectures. With this arrangement, we avoid having a more experienced lecturer skew the results in favor of either the KLA or traditional method.

### 4.2. Planning and Execution

There were several steps in planning the KLA study. The first step was to prepare for the lectures by deciding on the algorithms that would be most appropriate for the class. With the time given, we were able to plan for two class lectures in this KLA study. Pipeline algorithms were a part of the planned class curriculum, so we dedicated a class lecture to various parallel pipeline algorithms. There are also an abundance of parallelizable sorting algorithms which lend themselves well to KLA activities, so we decided to use the second class lecture to instruct the students on sorting algorithms. We then selected four different algorithms for each lecture. Our plan for this study was to split the class into two groups and have each lecturer spend half of a lecture instructing one group using KLA methods for two algorithms and the other half of the lecture instructing the other group of students on the other two algorithms using traditional methods. With this approach each group would receive both KLA and traditional lectures on a given lecture but on different topics and each lecturer would give instruction on two algorithms using both traditional and KLA methods. By organizing the study in such a way, we hoped to find good results on the effectiveness on KLA by comparing a lecturer's effectiveness using traditional methods versus their effectiveness using KLA methods.

The next step was to decide the method with which to divide the class so that we could alternate between the teaching of lessons using KLA and traditional methods. Because this KLA study took place over multiple days, we needed find a method that satisfied a few requirements. Each group of students needs to be of approximately the same size. We also needed a method of dividing the class that would be repeatable so that the students would be in the correct group for each lesson. We considered multiple methods before deciding to have the students split into groups based on whether their birthdays occur on an even or odd date. We labeled the students with odd birth dates as the orange group and provided pre-tests and post-tests on orange paper. We labeled the students with even birth dates as the white group and provided pre-tests and post-tests on white paper. This allowed us to use visual cues to easily differentiate between the two groups. This even and odd birth date approach provides several benefits. Because there are approximately the same number of odd and even dates in a year, we were able to theoretically get groups of about the same size. And because the dates of the students' birthdays remain static, the groups that they would divide into will remain the same through each of the sessions. Another important benefit is that this method is quick and simple for the students to follow, allowing more time for the lessons to occur.

With the planning complete, we could begin our study. In our first interaction with the students the week before the lectures, we briefed the students on our research and explained to them our goal, methodology and what they could expect from the upcoming lectures. We then obtained signed permission from each student in which they agreed to participate in the KLA study. Finally we provided a pre-test to the students to gauge the students' understanding of the algorithms chosen for each lecture so that we would have a baseline with which we could compare our final results.

For the first lecture, we divided the class according to our even-odd birth date methodology. The orange-odd group was asked to move to the smaller adjacent classroom where they received a lecture, containing KLA elements, on the prime number generation and pipeline insertion sort algorithms. The white-even group remained in the main classroom where they received a traditional lecture on a pipeline addition algorithm and pipeline system of linear equations algorithm. Once the lectures were complete, each group was asked to complete a post-test to measure their understanding on the algorithms in which they just received a lecture on. The instructors then exchanged classrooms where, in the main classroom, the students would receive a traditional lecture on pipeline prime number generation and pipeline insertion sort, and the students in the adjacent classroom would receive a KLA lecture on pipeline addition and pipeline system of linear equations. Once these lectures were completed another post-test was used to measure their understanding after receiving the second half of the pipeline lecture. In this way, the pipeline lecture was completed.

The second lecture occurred similar to our first lecture. The main difference was that the white-even group was asked to move to the smaller adjacent classroom where they received the KLA lectures. This white group received KLA lectures onthe topics of shear sort, parallel radix sort, odd-even sort, and parallel merge sort algorithms. The orange group received traditional lectures in the main classroom on the same topics. Two post-tests were given to students after each half of the lecture, one before the graduate instructors exchanged classrooms and another one at the end of the lectures, each test gauging the student's knowledge on the two algorithms they had learned in each half of the lecture.

## 5. Results

With the post-tests and pre-tests of the approximately 25 students in our single course, we can analyze the effectiveness of kinesthetic learning activities in a computer science classroom in comparison to more traditional teaching methods. We found in our study that the KLA methodis more effective at teaching students the selected algorithms and that they had a better understanding of the material after receiving a KLA lesson than they had from the traditional lesson.

Our results show that students who took the post-test after having been instructed with the KLA method scored 17.9% better than when they were instructed with a traditional method. The orange group improved 49.4% on their post-test over their baseline pre-test in parallel sorting when they learned from a traditional method but improved by 59.3% when they learned from a KLA lecture. Similarly the white group improved by 30.8% on their post-test over their pre-test after a traditional lecture and improved by 53.4% after a KLA lecture.

| | Pre-Test Odd (Orange) | | Post-test Odd (Orange) | |
|---|---|---|---|---|
| | Parallel Sorting | Pipeline | Traditional Parallel Sorting | KLA Pipeline |
| | 0 | 0 | 5 | 8 |
| | 0 | 0 | 7 | 9 |
| | 1 | 0 | 10 | 11 |
| | 1 | 0 | 10 | 15 |
| | 1 | 0 | 11 | 17 |
| | 1 | 0 | 12 | |
| | 2 | 1 | 12 | |
| | 3 | | 15 | |
| | | | 17 | |
| Average | 1.125 | 0.143 | 11 | 12 |
| Average Score (out of 20) | 0.05625 | 0.00715 | 0.55 | 0.6 |

*Table 5.1 Test Results for Odd-Orange Group*

| | Pre-Test Even (White) | | Post-test Even (White) | |
|---|---|---|---|---|
| | Parallel Sorting | Pipeline | KLA Parallel Sorting | Traditional Pipeline |
| | 0 | 0 | 5 | 0 |
| | 0 | 0 | 6 | 1 |
| | 1 | 0 | 10 | 2 |
| | 1 | 0 | 12 | 6 |
| | 2 | 0 | 12 | 6 |
| | 2 | 0 | 15 | 7 |
| | 2 | 0 | 15 | 10 |
| | 3 | 0 | 15 | 11 |
| | 3 | 0 | 15 | 11 |
| | 4 | 0 | 15 | 12 |
| | 5 | 5 | 16 | |
| | | | 17 | |
| Average | 2.09 | 0.455 | 12.75 | 6.6 |
| Average Score (out of 20) | 0.1045 | 0.0227 | 0.6375 | 0.33 |

*Table 5.2 Test Results for Even-White Group*

In the pre-test, a definite trend that can be seen is the low scores. This is to be expected as this is the first time many of the students have seen most of these algorithms. The scores of the post-test show an increase in understanding the materials after the lectures.

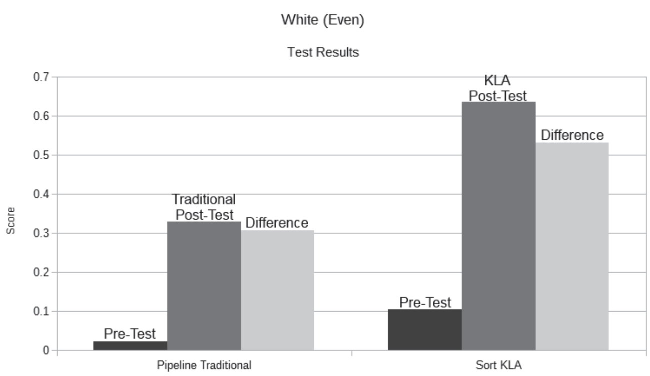



*Figure 5.3 Chart Showing Pre-Test and Post-Test Results for the Even-White Group*

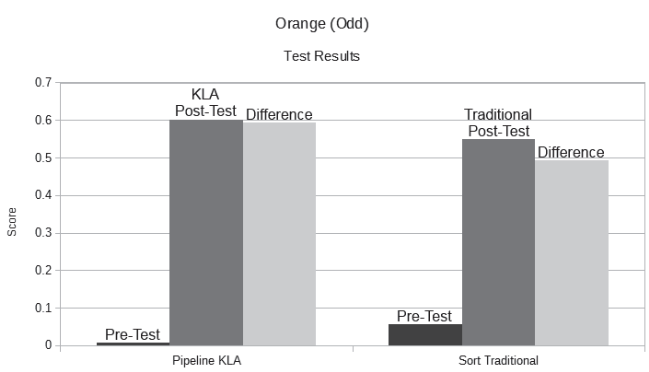



*Figure 5.4 Chart Showing Pre-Test and Post-test Results for the Odd-Orange Group*

In the two previous figures, you can see that the improvement between the pre-test and post-test is most pronounced after the KLA lectures. The greatest improvements to the students' scores came after they were instructed using the KLA method.

# 6. Conclusion and Future Work:

From the given figures and results, there appears to be a positive correlation in how well a student learns more from a KLA lectures in comparison to more traditional lectures. Using our methodology, we have shown this result using a single body of students over a single quarter. With the fact that we were able to run this test with two graduate student instructors individually teaching the same algorithms to each student group with both KLA and traditional methods, we can almost eliminate the effects that any difference in teaching skills between instructors might have on the results. With KLA we have a good teaching method that will keep students engaged and focused during class. Through the results of our experiment, we believe that KLA can be a powerful tool to teach students different topics of computer science and increase their understanding of the subject.

In further studies, we hope to be able to research what effects more experienced instructors may have on the score difference between traditional methods and KLA methods. More studies of KLA in the computer science field can only be an improvement to our current state of research as more experimental data on the effects of KLA on teaching students computer science methods will give more confidence to the effectiveness of KLA in computer science classrooms.

# REFERENCES

[1] Alraddady, Sara, Danny Luong, and G. Young. 2014. "A Study of Kinesthetic Learning Activities Effectiveness in Teaching Computer Algorithms." *Proceedings of the2014 International Conference on Frontiers in Education: Computer Science & Computer Engineering.* Las Vegas, NV: 400-406.

[2] Begel, Andrew, Daniel D. Garcia, and Steven A. Wolfman. 2004. "Kinesthetic Learning in the Classroom." *35th Proceedings of the Technical Symposium on Computer Science Education*. New York, NY: 183-184.

[3] Faust, Jennifer L., and Donald R. Paulson. 1998. "Active Learning in the College Classroom." *Journal on Excellence in College Teaching*, 9(2): 3-24.

[4] Fleming, Neil D. 2001. "Teaching and Learning Styles: VARK Strategies." Christchurch, New Zealand: Author.

[5] Gunion, Katherine. 2009. "FUNdamentals of CS: Designing and Evaluating Computer Science Activities for Kids." Victoria, BC, Canada: MS thesis, University of Victoria.

[6] Nguyen, Tammy. 2004. "Do Kinesthetic Strategies Influence Students" Achievement?" Pomona, CA: MS thesis, California State Polytechnic University, Pomona.

[7] Polovina, Simon. 2011. "About the Learning Pyramid." http://homepages.gold.ac.uk/polovina/learnpyramid/about.htm (accessed December 1, 2013).

[8] Sivilotti, Paolo and Scott M. Pike. 2007. "The Suitability of Kinesthetic Learning Activities for Teaching Distributed Algorithms." *38th Proceedings of the Technical Symposium on Computer Science Education*. New York, NY: 362-366.

[9] Wankat, Phillip C. 2002. "The Effective, Efficient Professor: Teaching Scholarship and Service." Boston, MA: Allyn and Bacon.

[10] Williams, Laurie, Robert R. Kessler, Ward Cunningham, and Ron Jeffries. 2000. "Strengthening the Case for Pair Programming." *IEEE Software*. 17(4): 19-25.

# Animation in an Introductory Computer Science Course

**M. R. Zargham**
Department of Computer Science
University of Dayton
Dayton, Oh USA
mzargham1@udayton.edu

**Abstract -** *This paper represents the type of projects that can be assigned in an introductory programing course based on Maya and Matlab. Students find such a course more interesting and creative than a typical computer science programing course. Students enjoy working in a group of three to develop something that they relate to. They write codes to do animation and are excited to do changes as necessary. They make every effort to do a good project while having fun; and that's what our goal is! Such a course will increase retention rate of students, in particular underrepresented group.*

**Keywords**: Introductory CS Course, Maya, Project Based learning.

## 1. Introduction

The Creative Media is a multidisciplinary, project-driven learning process that encourages students to develop problem solving and teamwork skills while fostering creativity and logic. The goal is to not only provide students with some "programming maturity," but to also engage them through working in small teams on existing projects related to their discipline and interest. Projects may include creative movies and games, mobile app developments, or avatars.

The course will be taught in a way that any student will be able to write their own code and create media. These are the first steps in exposing students to the roles of computing in today's society and understanding the foundations of computing. The only background that students are required to have is knowledge of how to turn on a computer. Step by step they will learn how to program by completing many simple and creative projects that are suited to their interests and needs.

Students will use "*Matlab*" as their first programming language to code the lab assignments and projects. At the same time, they will learn problem solving by using flowcharts. *Matlab* has a simple environment to develop programs, and in contrast to many conventional programming languages, it supports a visually responsive environment.

Although the proposed course will use *Matlab* as learning environment, the focus of the course will be on the computing fundamentals, preparing, and encouraging students to explore other languages and design environments. This course will not only be of benefit to both scientists and engineers, but also to any student working in a visual field, such as graphic design, painting, sculpture, architecture, film, video, illustration, and web design. Furthermore, this course will heighten the interest of students from diverse backgrounds and cultures by covering some basic functions of Maya after the first 4 weeks. As stated in *www.autodesk.com/products/autodesk-maya/overview*:

> "Maya® 3D animation software offers a comprehensive creative feature set for 3D computer animation, modeling, simulation, rendering, and compositing on a highly extensible production platform."

Once students have learned the basics of both Autodesk Maya and Maya Embedded Language (MEL), they can begin to manipulate and animate whatever they desire. They can create fully integrated and interactive 3D models, graphs, and more. Students who have learned the *Matlab* language can learn MEL's basic commands in a short period of time.

During the first four weeks of the class, students are involved in group projects. Each group is made of 2 to 3 students, mixing students of both strong and weak backgrounds and making every effort to have one minority student in each team when possible. Such groupings prevent strongly motivated students from being less engaged while providing more opportunity for weak students to succeed in the course. We will regularly monitor their progress during their collaborations and ensure that the work loads are equally distributed between the students. The students will also be involved in various training and social activities throughout the course.

***Targeted Student Participants:*** The course will attract students with diverse educational backgrounds to form teams to tackle various problems. There are many students at UD who have not declared a major in their freshman year. Many students in Engineering and Business schools take our introductory programming courses. Although the core of the course participants will be students with backgrounds in computer science, electrical/computer engineering and business, students from other disciplines such as mathematics, physics, chemistry, biology, communication, and arts are also encouraged to enroll and participate.

## 2. Assignments

Several assignments were assigned in the class including 3D modeling of an object based on their choice; using MEL (Maya Embedded Language) to do animation on a 3D model; moving objects by using music sound; Animating the movement of a planet; and finally to animate a dance scene. In the later project, students develop 3D modeling of their character(s), and also design the stage background and lighting. Figures 1 and 3 demonstrate some of these projects.

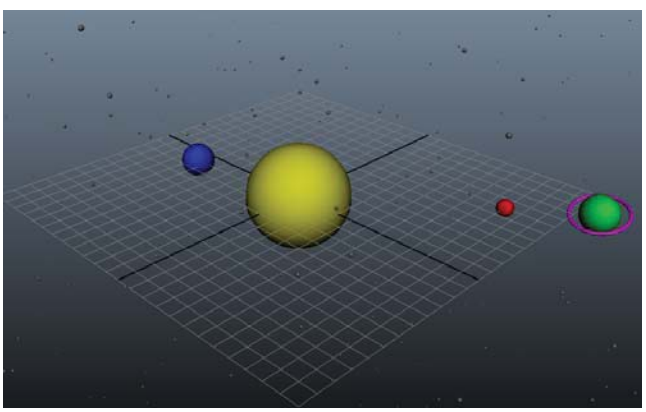
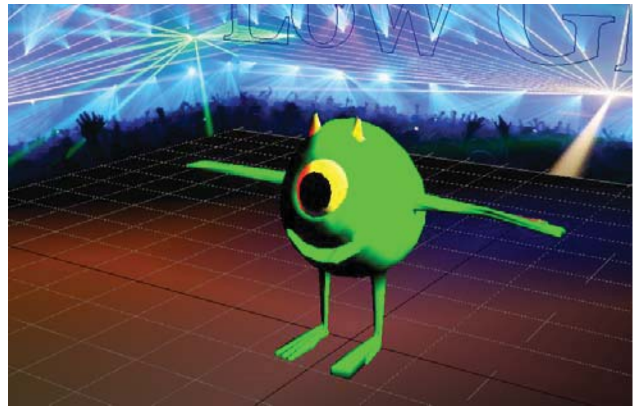


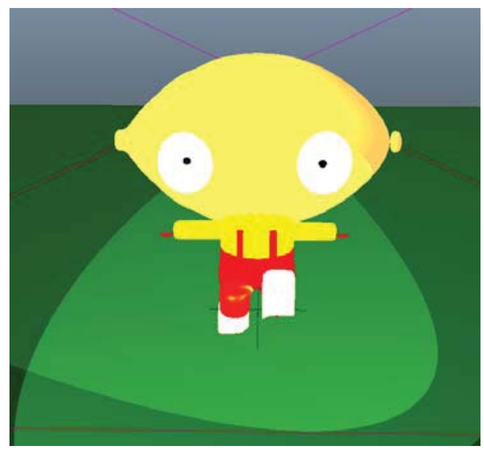
Figure 1: A Planet

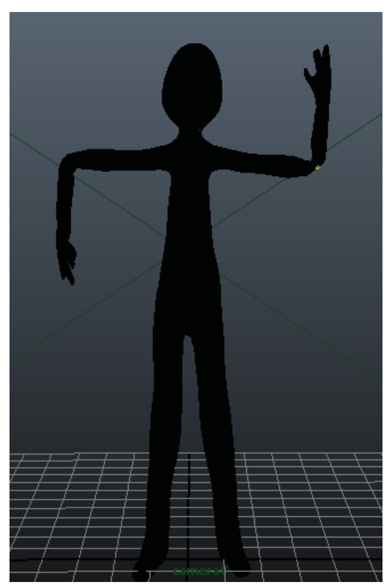
```
file -f -new;
int $max = 600;
int $min = 300;
int $noOfStars = rand ($min, $max);

float $satTx = 0;
float $satTy = 0;
float $satTx = 0;

polySphere -ch on -o on -r 1  -cuv 1 -n "earth" ;
move -r 6 0 -5 "earth";
polySphere -ch on -o on -n "sun" ;
scale -r 3 3 3 "sun";
polySphere -r .5 -n "mars";
move -r 9 0 -7 "mars";
sphere -r 1.1 -n "saturn";
move -r 13 0 -10 "saturn";

float $satTx = `getAttr saturn.translateX`;
float $satTy = `getAttr saturn.translateY`;
float $satTz = `getAttr saturn.translateZ`;

polyTorus -r 1.6 -sr 0.1 -n "ring";
rotate -r 15 0 0 "ring";
//move -r 13 0 -10 "ring";
setAttr ring.translateX $satTx;
setAttr ring.translateY $satTy;
setAttr ring.translateZ $satTz;

int $i;
for ($i=0; $i < $noOfStars ; $i++){
   float $radius = rand(0.01, 0.11);
   float $tX = rand(-18, 18);
   float $tY = rand(-18, 18);
   float $tZ = rand(-18, 18);
   polyPrimitive -r $radius;

   move -r $tX $tY $tZ;
   }

//createAndAssignShader mars_Color "";


shadingNode -asShader lambert;
// Result: lambert2 //
sets -renderable true -noSurfaceShader true -empty -name lambert2SG;
// Result: lambert2SG //
connectAttr -f lambert2.outColor lambert2SG.surfaceShader;
// Result: Connected lambert2.outColor to lambert2SG.surfaceShader. //
select -r lambert2 ;
//showEditor lambert2;//
setAttr "lambert2.color" -type double3 1 0 0 ;
select -r mars;
sets -e -forceElement lambert2SG;

shadingNode -asShader lambert;
sets -renderable true -noSurfaceShader true -empty -name lambert3SG;
connectAttr -f lambert3.outColor lambert3SG.surfaceShader;
select -r lambert3 ;
setAttr "lambert3.color" -type double3 0 0 1 ;
select -r earth;
sets -e -forceElement lambert3SG;

shadingNode -asShader lambert;
sets -renderable true -noSurfaceShader true -empty -name lambert4SG;
connectAttr -f lambert4.outColor lambert4SG.surfaceShader;
select -r lambert4 ;
```

```
setAttr "lambert4.color" -type double3 1 1 0 ;
select -r sun;
sets -e -forceElement lambert4SG;

shadingNode -asShader lambert;
sets -renderable true -noSurfaceShader true -empty -name lambert5SG;
connectAttr -f lambert5.outColor lambert5SG.surfaceShader;
select -r lambert5 ;
setAttr "lambert5.color" -type double3 0 1 0 ;
select -r saturn;
sets -e -forceElement lambert5SG;


shadingNode -asShader lambert;
sets -renderable true -noSurfaceShader true -empty -name lambert6SG;
connectAttr -f lambert6.outColor lambert6SG.surfaceShader;
select -r lambert6 ;
setAttr "lambert6.color" -type double3 1 0 1 ;
select -r ring;
sets -e -forceElement lambert6SG;

//shadingNode -asShader mars_Color;//
// Result: lambert5 //
//sets -renderable true -noSurfaceShader true -empty -name mars;//
// Result: lambert5SG //
//connectAttr -f mars_Color.outColor mars.surfaceShader;    };//

//setAttr "mars_Color.color" -type double3 1 0 0.151964 ;//

select -r earth ;
setKeyframe "earth.ry";
// Result: 1 //

currentTime 240 ;
rotate -r -eu 0 358.019944 0 ;
setKeyframe "earth.ry";
// Result: 1 //

playbackOptions -min 1 -max 240 ;
// Press the ESC key to stop playback. //
currentTime 1 ;

select -r earth ;
doGroup 0 1 1;
rename "group1" "earth_orbit";
// Result: earth_orbit //

select -r earth_orbit ;

currentTime 1 ;
setKeyframe "earth_orbit.ry";
// Result: 1 //

currentTime 240 ;
rotate -r -eu 0 720.0 0 ;
setKeyframe "earth_orbit.ry";
// Result: 1 //
playbackOptions -min 1 -max 240 ;
currentTime 1 ;
```

Figure 2: MEL code for the planet in Figure 1.







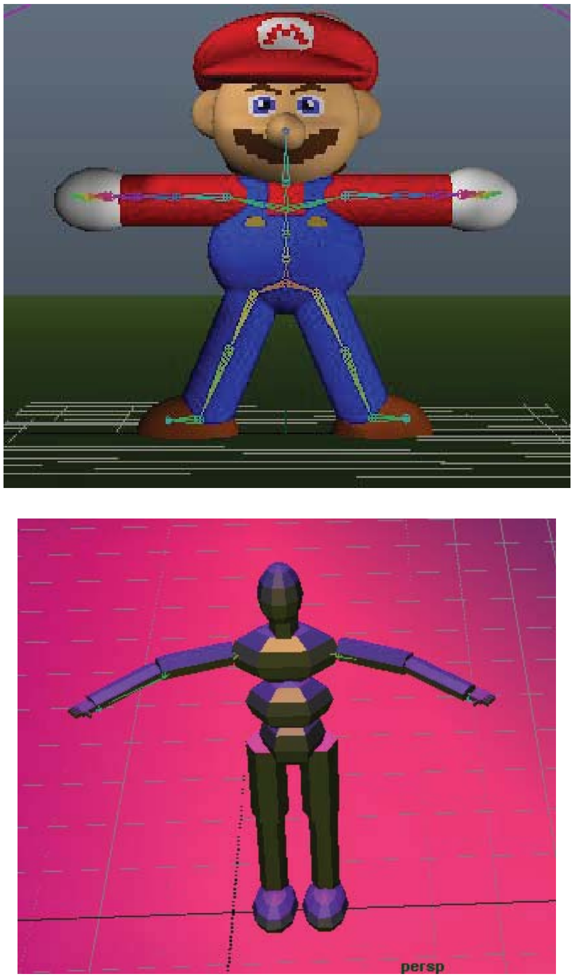



Figure 3: Different students project for the dance scene.

In regards to Matlab, students designed many flowcharting for adding the elements of a one dimensional array; flowcharting for adding the elements of each column of a two dimensional matrix. They also did several projects including Tic Tac Toe. There are other projects that can be used for such a course; some of them are listed below.

***Body Mass Index (BMI)*** **(all teams).** This app accepts the weight (in ponds) and height (in feet) from users and displays their BMI. It first converts the entered weight and height to kilograms and meters, respectively. Then calculates BMI by dividing weight by height squared (i.e. BMI=Kg/m$^2$). Based on the BMI value, the app displays one of the appropriate below terms for the user.

| Adults | Women | Men |
|---|---|---|
| Anorexia | <17.5 | |
| Underweight | <19.1 | <20.7 |
| In Normal Range | 19.1-25.8 | 20.7-26.4 |
| Marginally Overweight | 25.8-27.3 | 26.4-27.8 |
| Overweight | 27.3-32.3 | 27.8-31.1 |
| Very Overweight or Obese | >32.3 | >31.1 |
| Severely Obese | 35-40 | |
| Morbidly Obese | 40-50 | |
| Super Obese | 50-60 | |

***Campus Tour*** **(two teams to three teams).** In every college, there is a new student admission office that often provides campus tour to prospective students and their parents. The goal of this project is to do this task by developing a self-tour guided app that displays a list of the five nearest building using GPS technology. The distance to each building will be displayed as well as each building name. A video about each building is placed next to the name in case the user wishes to see which building the name is referring to and also to know the building's history and the units it includes. The app also shows the current position of the visitor at any moment on the campus map.

Given the fact that implementation of this project is hard for many beginners, we provide students with a basic sample app and ask each team to improve the code.

***Survivor game***. There are two types of Orbs, blue and red. There are ten of each kind. Each kind of Orbs needs to hit another Orb in order to survive. The attacked Orb will die. An Orb can hit another Orb by hitting its sharp edge to the round part of the other Orb. The stronger Orbs will survive!

Each Orb is a neural net. They are capable of self-improvement in order to become stronger. Each Orb can see the location and direction of the closest Orb of the different type. Each Orb is able to do four actions, move forward, turn left, turn right, and hit.

***Vacuum Robot.*** In this project you will program a simulated robot to navigate through its environment. The robot senses its environment by using bump sensors (in case of hitting objects or walls), global position, dirt sensors. With the data from sensors your program should decide an appropriate move for the robot to execute.

***Maze Robot.*** In this project you will program a robot to find its way through a maze going from a starting point to an end point. A maze is a grid of square cells where each cell is a wall or an empty path. The robot should travel through empty cells to reach to its destination. At each state of the movement, the robot can see (sense) its four neighboring cells (South, East, North, and West) and based on these cells it should decide its move direction.

***Gender Classification*** This project classifies male and females based on their full-body pictures. Students will address the following specific tasks: 1) Constructing a data set by asking many volunteers to have their pictures taken from a front angle, full-body shots. They will take full-body shots from a specific distance with the subject standing completely erect against a white wall. White background area is used to simplify feature extraction in later stages; 2) Learning MATLAB; 3) Selecting important features (such as hair, shoulders, height, body mass, etc.) and extracting them from images by using MATLAB. The major difficulty here will be to eliminate noise as much as possible and reduce the number of objects in the picture in order to have more accurate measurements of selected features of a given person. 4) Once they have decided which features to use and how to extract them, their next step will be to train a model to distinguish gender.

***Zoo Tour.*** In Cincinnati, Ohio there is a beautiful Zoo that attract many tourist and local people. Unfortunately, there is nobody to offer a comprehensive tour to visitors. The goal of the first phase of this project is to do this task by developing a self-tour guided app that displays a list of the five nearest cages using GPS technology. The distance to each cage will be displayed as well as each cage name. A video about each cage is placed next to the name in case the user wishes to see which cage the name is referring to and also to know detail information about the animals that cage includes. The app also shows the current position of the visitor at any moment on the Zoo map. However, in practice, it is possible that the Zoo replace the animals in a cage without updating the app's GPS coordinates. To address this problem, in the second phase of this project (perhaps in another course), the students modify the app by including machine learning capability. If the user has doubt about the residents of a cage, he/she may take a picture the resident animal in order to be recognized by the app.

# 3. Conclusions

In this paper, we have described several projects that can be assigned in an introductory computer science course to retain students. We have implemented such a course and preliminary results indicate that we have lost only two of the freshmen students during the academic year 2014-15. The students have gained more confidence and are performing well in the upper classes.

# 4. References

[1] Blum, L. 2001, “Women in Computer Science: The Carnegie Mellon Experience”, women@scs2.2., to appear in The Future of the University: The University of the Future.

[2] Klawe, M, 2013, “Increasing Female Participation in Computing: The Harvey Mudd College Story,” IEEE Computer, pp. 56-58, 2013.

# Development of an Automated Information System: A Students' Learning Perspective

**Sandip C. Patel**
Information Science and Systems, Morgan State University, Baltimore, Maryland, USA

**Abstract** - *Automated information systems have been getting much attention since the 1990s. However, the students at the undergraduate level find developing the software for automated information systems difficult. In this research, we take an application of the decision support system (DSS), a type of information system, and show students how the specifications, design, algorithms, and code are created for an automated DSS for irrigation scheduling. We also illustrate how the DSS was implemented in a field. The commercially available DSS demand much information from farmers and give output in terms of irrigation water quantity which farmers must measure. The DSS created in this research help farmer make irrigation scheduling without them having to input too much details into the software or measure irrigation water quantity.*

**Keywords:** Teaching software development; code development method; information system development; systems development pedagogy

## 1 Introduction

Proper irrigation water management is necessary for the optimum crop yield and reducing costs. This management is the technique for applying water according to crop needs, which depends on amount that can be stored effectively in the plant root-zone. Irrigation scheduling [2, 14] is used for determining the timing, frequency, or quantity of water released for irrigation. Irrigation scheduling heavily affects crop yield and operating costs. Irrigation scheduling determines when and how much water is released into a farm. Saving water and energy, irrigation scheduling aims to maximize irrigation efficiencies by applying the exact amount of water needed by the plated crops in a farm. Proper irrigation scheduling helps maximize crop yields, reduce watering and fertigation costs, improves crop quality, reduces chances of water logging, helps control soil salinity [8], and saves water.

The irrigation scheduling methods are based on water-balancing, soil moisture measurement [12], using computer models, or manual chart keeping [13]. The manual scheduling methods do not result in optimum irrigation scheduling and hence researchers have tired automated irrigation scheduling methods. The automated systems that are commercially available have some parts of the data transfer from field to the decision making process is automated but the other parts are left manual. For example, some hardware companies provide solution to have the filed data collected by sensors [1, 3, 4, 6, 7] to smart phones or to the Internet but the system do not provide support for turning the irrigation system to on or off using electronic media. In addition to requiring the farmers to do so manually, the decision making to turn irrigation on and off is left at the discretion of the farmer. Thus, some wireless systems provide support for partial monitoring or partial control but not completely. Some companies offer the services so that the farmer can monitor the field situation via Internet at additional cost but the user cannot make any changes or enhancement to what the system provides.

Soil moisture deficit is the amount of water required to raise the water content of the root zone to the maximum water a filed can hold, which is referred as field capacity (FC). Soil water is stored in the soil pores but all water is not available for plant roots. Soil moisture tension represents how tightly water sticks to the soil and is measured in pressure units of centibars. For example, soil moisture tension is very low at 100% FC since the soil not holding the water very tightly (http://cals.arizona.edu/pubs/water/az1220/). At the higher soil tension, it becomes increasingly difficult for plants to extract water from soil till the moisture level reaches the permanent wilting point (PWP) at which point plant cannot extract any water from soil. PWP is different for different types of plant and it also depend on the type of soil.

Figure 1 shows the proposed solution where we provide the irrigation scheduling and monitoring via the Internet and cellular phone devices. The layout provides end-to-end solution from the field data to the Internet and the users have choice of manipulating the data per their need without any additional cost. The irrigation control is automated where the DSS would make the decision that is optimum. This DSS could also be edited and changed per changing needs.

As illustrated in Figure 1, the field data from the soil moisture sensors are sent to a data logger which is located on the field. The data logger sends the data collected from several sensors to a personal computer (PC) using wireless technology such as GSM or GPRS, which uses radio waves to transmit the data. Application software that accompanies field sensors is required to convert these data into a readable format such as an Excel spreadsheet. The program that we created provides decision support system (DSS) to turn the irrigation on or off. This decision can be transmitted to the Internet for remote monitoring [9, 15]. The decision can also be transmitted to field control system to start or stop drip irrigation system [11]. A mobile phone can also be used as a control and monitoring device for the irrigation system.

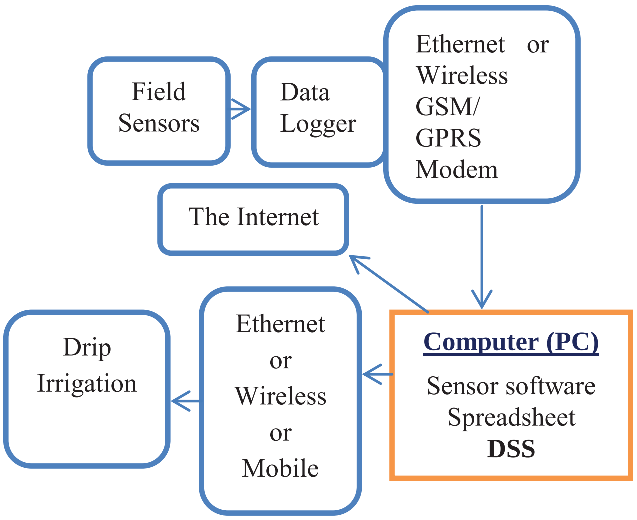



**Figure 1 Layout of the automated information system**

In the proposed research, we implement soil moisture sensors to collect soil moisture and temperature data and automatize irrigation with the optimum water amount and time of irrigation through an automated drip irrigation/fertigation system. Using an integrated approach to monitor soil moisture, electrical conductivity (EC), and sodicity (pH value), and supplying these values to a rule-based decision support program would determine an optimum amount and time of irrigation in conjunction with an automated drip irrigation system.

In this paper, we show how we developed a DSS that help farmer make irrigation scheduling. Sensor-based irrigation scheduling helps in continuous moisture monitoring and control at the effective root zone, which is the depth of soil at which moisture control is most effective for a plant type. It provides continuous-reading. The newest developments in the continuous-reading systems utilize electric-powered sensors of some type, which includes a data logger getting its data from tensiometers, granular blocks, or FDR/TDR ("Graphical Irrigation Scheduling," http://cwi.csufresno.edu/wateright/Sched2.asp).

# 2 DSS Development Method

We used the software development life cycle (SDLC) [10] method to develop the software. The phases of SDLC are development of requirements and specifications, software design, development or coding, testing, validation & Verification and then implementation. We show how these phases were applied to the automated DSS development in the following subsections.

## 2.1 Specifications and Requirements

The following general requirements were set by discussing the DSS with the prospective users, and by literature study:

- Programming language and user Interface: Use a simple and user friendly programming language.
- Visual component: We needed GUI-based application software.
- Integration with irrigation system: We needed that the system takes input from the field sensors and the output is in the format that can be fed to the irrigation system control to turn it on or off.
- Simplicity: We wanted our model to be simple enough so that the associated code is not too complicated to be understood, modified, or enhanced by the people other than those who developed it.
- Flexibility: We wanted our model to be flexible so that different plants, root-depth, mulching, and so forth could be accounted for.
- User-friendly and easy interface: The DSS should have an easy and comfortable look and feel.
- Portability: The DSS should be able to run on most computer and operating systems.
- Scalability: We wanted the DSS to be scalable so that we can add different functionalities or enhance the DSS for different scenarios, for example, to examine effects of mulching.

The following summarizes the program specifications that were set forward after discussions and interview with Junagadh Agricultural University (JAU), Junagadh, India faculty.

Using the data file created by Virtual Company's sensors, the developed program shall:

- Calculate the average soil-moisture tension in centibars at the end of the day. That is, the daily averages will be used; neither hourly nor part-daily will be used.
- Calculate the average soil-moisture tension in centibars for two columns for each sensor since one column represents a sensor at a shallow depth and the other at the deeper depth for the same mulching type. For example, if there are eight columns for the tension, the program calculates four daily averages: The daily average of the first two columns, the next two columns, and so forth.
- Use the above averages (of two columns each) to find the moisture content (MC) using a curve/model that will be supplied later.

- Make decisions using conditions. Initially, use the same and simple condition to start irrigating at 30% MC and stop irrigating at 50% MC for each of four averages, which represent four types of mulching: silver & black plastic mulch, black mulch, wheat straw, and no mulch. Later, these conditions might be modified to be more refined. But initially, four decisions such as below will be made:
- If MC falls below 30% for silver & black plastic mulch, start irrigating. When MC reaches to 50%, stop irrigating.
- If MC falls below 30% for black mulch, start irrigating. When MC reaches to 50%, stop irrigating.
- If MC falls below 30% for wheat straw, start irrigating. When MC reaches to 50%, stop irrigating.
- If MC falls below 30% for no mulch, start irrigating. When MC reaches to 50%, stop irrigating.

Currently, do not take the temperature into account but can be used in the future.

## 2.2 Design

During the design phase, the requirements and the specifications are translated into design features. In addition, functions and operations of the software may be described during the design phase (Patel et al., 2014). Such descriptions include screen layouts, business rules, and process description and diagrams. The output of this stage describes the new system as a collection of modules or subsystems. After considering the general requirements and specifications listed above, the different types of DSS techniques, and best practices in programming, we found that the automated DSS should have these design features: good visual component, simplicity, flexibility, data collection at granular time periods, be able to present aggregated data, portability, scalability, user-friendliness, convenience of use, and capability to handle multiple mulching, multiple crops, multiple root-depths.

### 2.2.1 Data structures

Using Microsoft's Interoperability library for Excel, the input spreadsheet was read by the DSS as Visual Basic's "Application" object. Variables were used for 'Total no of rows in the spreadsheet, 'Total no of columns in the spreadsheet and others to cell manipulations. An array was used called data_spreadsheet so that the program can read any size of excel spread. Since average values are used, an integer variable is used to track the number of values. This variable is used to divide the total number of values to arrive at the average value.

**Input file**: Each column of the input file represent data from each sensor. Typically, two sensors (and hence two columns) are placed for each mulching type as shown in Table 2 in Appendix. Each row lists data for different date and time. The first column of this table shows the name of the project site. The second column shows the date and time of the sensor measurement. The third column shows the soil temperature which is not considered in the current version of DSS but will be used in the future development. The rest of the columns, except the last column, show the soil moisture in centibars. Each mulching-type has two sensors installed. Thus, for example, columns four and five represent silver & black plastic mulching. The last mulching type has only sensor installed. The last column shows battery voltage, which is currently not used by the DSS.

### 2.2.2 Screen Objects

Screen was kept simple by keeping only two GUI objects including a button, a text box, and a list box. The button is used for start running the DSS program, the text box was used for inputting the spreadsheet file's location and name, and the list box was used for displaying the centibar values. The purpose of the DSS was to pass the "start irrigation" and "stop irrigation" decisions to the irrigation system, and not to a person. Hence the screen output of the decision was used as a supplemental output. The data in the list box could be primarily used for further analysis of the data or debugging the system. These data do not have direct application for the irrigation system's operation.

The text box, used for inputting the spreadsheet of data generated by sensors, contains the default file name and its directory location. Therefore, the user can simply start running the program without having to enter the file name and path or can modify the file name, or can modify the path.

### 2.2.3 Procedures

All code is placed under button_click procedure. Note that Procedures are known as private subroutine in VB.NET. In addition, three subroutines were created that are used by the button_click sub routine. The sensors_for_same_date is the major sub called by button_click, which takes the row numbers from the spreasheets and adds appropriate readis for the same day. Data from the same type of mulching is passed to this sub routine in order to calculate the daily averages of all the readings from two sensors in a day.

**Function: run_model():**

This function converts the soil-tension into moisture content. Depending on the threshold set by user, the module makes the decision to turn the irrigation on or off. In addition to displaying the decision on the screen, this routine sends a command to the irrigation control system, specifically, remoter terminal unit, to turn the irrigation on or off. Although the development of the DSS is complete, the integration of the DSS is still to be completed. This integration will enable the DSS to send signals to RTU to turn the irrigation on or off.

## 2.3 Coding for the irrigation scheduler

In the coding phase of SDLC, various decisions regarding writing the code are taken, detailed pseudocode is generated, and then actual lines of code are written. We decided to use Visual Basic.NET (VB) for its simplicity and visual interface. Configured VB for Web development environment configured the VB reference libraries to run the program. This program required that Microsoft Excel Object Library was added to read the Excel file.

### 2.3.1 Pseudocode

In sensors_for_same_date procedure (VB.NET sub routine), three synerios are considered. First, the data from the first row containing numbers to the last-minus-two rows is considered. Note that the first two rows contain column headings and hence the data starts from the third row. Then, special case for the last two rows so that the array is not out of bounce. This routine takes tow column numbers as input since each mulching has two sensors. One sensor is at shallow depth (15 centimeters) and the other is deeper (30 centimeters). Track variable keeps count of how many readings were taken in a day. This is used to calculate the average daily centibars.

Three cases are considered for the last two rows. (1) The data in the last two rows are from the same date as the rows above them. (2) The data in the last two rows are from a different date, and (3) the data in the last row is from a different but the one above it is from the same date as those above it.

**Table 1: Possible scenarios for the last few rows of the input spreadsheet**

| Scenario 1 | Scenario 2 | Scenario 3 |
|---|---|---|
| : | : | : |
| 06/06/ 2015 | 06/06/ 2015 | 06/06/ 2015 |
| 06/06/ 2015 | 06/06/ 2015 | 06/06/ 2015 |
| 06/06/ 2015 | 06/06/ 2015 | 06/06/ 2015 |
| 06/06/ 2015 | 06/06/ 2015 | 06/06/ 2015 |
| 06/06/ 2015 | 06/06/ 2015 | 06/07/ 2015 |
| 06/06/ 2015 | 06/07/ 2015 | 06/07/ 2015 |

The exact pseudocode is as following:

**Procedure: button_click():**

Try to open the input spreadsheet by using the user-supplied file name and location. (There is a default value that can be overwritten by the user.)

If the file can be found at the supplied location, read it as Visual Basic's "application" object.

Loop over the entire sheet, count the total cells, rows, and columns and display these values on screen.

If the file is not found, alert the user to input the correct file.

Call sensors_for_same_date() with the supplied sets of columns (which represent sensors).

Close the input file.

**Procedure: sensors_for_same_date():**

For the third row to the last-minus-one row

If the current row and the next row have the same date (NOTE: we are not considering the last row here. Table 1 scenario 1 and 2 possible)

Perform following for the first column:

Add the centibar value to FirstCol_Total and increase the track variable

A special case for the last two rows so that the array is not out of bounce:

If the last two rows of the table have the same date (Table 1 scenario 1 and 3 possible. But combined with previous If, it is scenario 1)

1. Perform the following for the first column:

   Add the centibar value of the last row to the above total value (FirstCol_Total); and increase the track variable

   Calculate the average value by dividing by the track variable and display it in the list box. Reset FirstCol_Total to zero.

2. Perform the same two above steps for the second column

3. Calculate the average of the first and the second column and display it in the list box.

Call run_model() to make the irrigation decision by supplying it the average value

End If

Else (The current row and the next row has different dates. NOTE: When it comes to the last two rows, we are not considering the last row here. Second-from-the-last row has a different date. Table 1 scenario 3)

1. Perform the following for the first column:

Calculate and display all values for the current date before moving to the next date. That is,

Add the centibar value to the total value (FirstCol_Total); and increase the track variable

Calculate the average value by dividing by the track variable and display it in the list box. Reset FirstCol_Total to zero.

2. Perform following for the second column:

Repeat the above steps for the second column

3. Calculate the average of the first and the second column and display it in the listbox.

Call run_model() to make the irrigation decision by supplying it the average value

End If

Next

Another special case where the last two rows have different dates (Table 1 scenario 2).

Get the values of both the columns of the last two rows and display the average

**Procedure: run_model()**

Read the supplied soil-tension value.

Convert the soil-tension into moisture content using appropriate equation.

Depending on the threshold set by user, the decide to turn the irrigation on or off.

Display the decision on screen.

Send a message to the irrigation remote terminal unit of the decision. (Currently not implemented)

## 2.4 DSS testing, validation & verification

During the testing phase, that code that was written is tested using various methods such a as unit testing and integration testing. Unit testing and integration testing of DSS was completed successfully. That is, when a sample input file was supplied, the DSS gave calculated the irrigation scheduling correctly verifying the integration testing. We also analyzed each module of DSS for unit testing. Input and output to each module was tested. Each data point was passed to the code modules and tallied with the results calculated manually. In addition, debugging code was inserted throughout the code which verified that each value at modular, routine, and functional levels was calculated correctly.

### 2.4.1 DSS execution and results

The actual field data was collected using the sensors from Virtual Electronics by the Department of Renewable Energy and Rural Engineering located at the College of Agricultural Engineering and Technology, Junagadh Agricultural University, Junagadh, India. This data were passed to the DSS. The results of irrigation scheduling were also done manually to tally the DSS results. Sensor data for various type of mulching were collected including silver & black plastic mulch, wheat straw, and crop without any mulching.

## 2.5 Implementation

We examined several field sensors such as that from Irrometer [5] which uses data logger Watermark Model 975 (replaced 950R) with wireless connection capacity. Eventually, we used Digital Soil Moisture Recorders from Virtual Electronics Company, which uses the GSM/GPRS modem to send the field data to a Web portal. (http://virtualweb.co.in/yahoo_site_admin/assets/docs/LF_Soil_Moisture_Recorder.177202954.pdf).

# 3 Findings and Conclusion

We successfully ran the DSS program on a real data file created by Virtual Company's sensors. There is no specific hardware requirement for the developed DSS to run. As mentioned above, we implemented the DSS software on an actual farm at Junagadh Agriculture University (JAU) and found it to be user friendly and easy to use. We found the DSS to meet the requirements that were specified during the initial stage of the SDLC. The software fulfilled the functionalities it was designed for.

In conclusion, we illustrated how automated decision support system could be created to undergraduate students in this paper. We showed how we completed all the steps of the SDLC. We identified problems with existing software, came up with specifications of new software, then designed, developed, and implemented the software.

## Acknowledgement

This research was completed with the help of Fulbright's core Academic and Professional Excellence award under US Scholar program for 2014-15.

# 4 References

[1] Akyildiz, I.F., Su, W., Sankarasubramaniam, Y. and Cayirci E. (2002). Wireless Sensor Networks: a Survey, *Computer Networks*, 38(4), 393-422.

[2] Alves, I., and Pereira, L.S., (2000). Non-Water-Stressed Baselines for Irrigation Scheduling With Infrared Thermometers: A New Approach, *Irrigation Science* 19, 101–106.

[3] Aqeel-ur-Rehman, Zafar Abbasi, A., Islam, and N., Shaikh, Z. A. (2011). A Review of Wireless Sensors and Networks' Applications in Agriculture *Computer Standards & Inter*faces, Available online from Elsevier.

[4] Garcia-Sanchez,A. J., Garcia-Sanchez, F., and Garcia-Haro, J. (2011). Wireless Sensor Network Deployment for Integrating Video-Surveillance and Data-Monitoring In Precision Agriculture over Distributed Crops, *Computers and Electronics in Agriculture*, 75(2), 288-303.

[5] Irrometer: http://www.irrometer.com/basics.html

[6] Jongschaap R.E. (2006). Integrating Crop Growth Simulation and Remote Sensing to Improve Re-Source Use Efficiency in Farming Systems. A Ph. D. Thesis submitted to Wageningen University,Wageningen, the Netherlands.

[7] Lillesand T.M., and R.W., Kiefer (2000) *Remote Sensing and Image Interpretation*, 4th ed. Wiley, New York.

[8] Minhas, P.S. and Tyagi, N.K. (1998). Guidelines for Irrigation with Saline and Alkali Waters. *Bulletin No. 1/98*, Central Soil Salinity Research Institute, Karnal, India.

[9] Mishra, P, Tiwari, and Gontia, N.K. (2010). Irrigation Water Demand and Supply Analysis in the Command Area Using Remote Sensing and GIS, *Water Resource Management,* 24, 1399–1414.

[10] Patel, S., Bhatt, G., & Emdad, A. (2014). Development of a Simulator for Complex Systems: A Students' Learning Perspective. American Journal of Information Technology, 4 (1), 33-49.

[11] Patel, V.B. (1999) Drip Irrigation: Using Water Sensibly. *The Hindu survey of Indian agriculture*, 165–168.

[12] Ray, S.S, and Dadhwal, V.K. (2000). Estimation of Evapotranspiration of Irrigation Command Area Using Remote Sensing and GIS. *Agricultural Water Management*, 49,239–249.

[13] Subbaiah, R. (2013). A Review of Models for Predicting Soil Water Dynamics during Trickle Irrigation. *Irrigation Science* 31(3), 225-258.

[14] Thompson, A. L., J. C. Henggeler, D. L. Bockhold, and K. A. Sudduth. 2002. Comparison of eight irrigation scheduling tools on soybean and cotton. *Journal of International Water and Irrigation.* 22(3): 24-30.

[15] Zade, M., Panigrahy, S. and Parihar, J. S. (2005). Spatial Curve Number Map for Major Basins of India Using Remote Sensing and GIS. *Proceeding National Conference on Sustainable Management of Water Resources*, held during 28-29 January, 2005, at CAET, Juanagadh Agriculture University, Junagadh, Gujarat, India, 223-226.

## Appendix

**Table 2: Sample file generated by data logger and supplied as input to the DSS system**

| Site Ref. | Date & Time | Soil Temp. | Soil Moist -1 | Soil Moist-2 | Soil Moist -3 | Soil Moist -4 | Soil Moist -5 | Soil Moist -6 | Soil Moist -7 | Batt. Voltage |
|---|---|---|---|---|---|---|---|---|---|---|
| | DD/MM/YYYY - HH:MM | °C | Centi-bars | Centi-bars | Centi-bars | Centi-bars | Centi-bars | Centi-bars | Centi-bars | Volts |
| A | 27/06/ 2012 - 12:00 | 32.8 | 30.5 | 154 | 13 | 200 | 200 | 71 | 138 | 200 |
| A | 27/06/ 2012 - 15:00 | 34.1 | 30.5 | 141 | 13 | 200 | 200 | 62 | 127 | 200 |
| A | 27/06/ 2012 - 18:00 | 35 | 30.6 | 127 | 13 | 200 | 200 | 54 | 114 | 200 |
| A | 27/06/ 2012 - 21:00 | 34.8 | 30.5 | 111 | 12 | 200 | 200 | 49 | 101 | 200 |
| A | 28/06/ 2012 - 00:00 | 34 | 30.3 | 94 | 12 | 200 | 200 | 43 | 95 | 200 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| A | 28/06/ 2012 - 03:00 | 33.3 | 30.1 | 83 | 12 | 200 | 200 | 40 | 82 | 200 |
| A | 28/06/ 2012 - 06:00 | 32.7 | 29.9 | 73 | 12 | 200 | 200 | 35 | 72 | 200 |
| A | 28/06/ 2012 - 09:00 | 32.3 | 29.7 | 64 | 13 | 200 | 200 | 32 | 62 | 200 |
| A | 28/06/ 2012 - 12:00 | 33.1 | 29.6 | 58 | 13 | 13 | 200 | 28 | 53 | 0 |
| A | 28/06/ 2012 - 15:00 | 34.5 | 29.9 | 52 | 13 | 14 | 200 | 26 | 45 | 0 |
| A | 28/06/ 2012 - 18:00 | 35.5 | 30.3 | 47 | 13 | 16 | 163 | 23 | 39 | 0 |
| A | 28/06/ 2012 - 21:00 | 35.2 | 30.5 | 45 | 13 | 17 | 158 | 21 | 34 | 0 |
| A | 29/06/ 2012 - 00:00 | 34.2 | 30.3 | 42 | 13 | 18 | 189 | 20 | 31 | 0 |

# Implementation of Best Practices Using a MOOC at a Historically Black University

**Velma Latson[1], Lethia Jackson[1], Courtney Lamar[1] and Monika Gross[2]**
[1]Department of Computer Science, Bowie State University, Bowie, MD, USA
[2]Undergraduate Honors Program, Bowie State University, Bowie, MD, USA

**Abstract** - *Bowie State University, a Historically Black University, in collaboration with the University System of Maryland and Ithaka S+R participated in an extended research study supported by a grant from the Bill and Melinda Gates Foundation. The original research study produced best practices for using MOOCs to teach programming to undergraduate students with very little or no programming experience. This paper discusses the extended research study as a true adaptive learning experience that utilizes the best practices as revealed in the original research study. The extended research study provides an open approach for hybrid learning as it enhances the use of technology and teaching in today's pedagogical learning landscape. The implementation of the best practices incorporated into the course's redesign proved to be successful for using the MOOC to teach Python programming language to non-STEM majors.*

**Keywords:** MOOC, computer science education, blended instruction, hybrid instruction

## 1 Introduction

Computing Principles and Technologies (COSC 111) is an undergraduate, general education course. The purpose of the course is to introduce non-STEM majors to programming, hardware, web development, technology, networking and security [6]. During summer 2013, COSC 111 was redesigned using a MOOC in a research study conducted by the University System of Maryland (USM) and Ithaka S+R. The redesign included an eight-week programming module which incorporates a Python programming MOOC developed by Rice University using the Coursera platform [5][6]. The redesigned COSC 111 course was taught fall 2013. A comparative research study was conducted using nine sections of COSC 111. Lessons learned in the original research study showed that using a MOOC as the primary delivery method in teaching programming concepts was a key factor to success. Following is the list of best practices integrated in the extended research study discovered in the original research study [9] [10]:

1) incorporate algorithms and logic exercises prior to writing a program
2) allow students sufficient time to learn how to navigate through the Learning Management System (LMS)
3) provide professional development for faculty when introducing new content
4) provide professional development for faculty with little knowledge using technology
5) offer peer to peer tutoring required weekly
6) provide on-site technical assistance for faculty and students
7) rotate the course material to offer the MOOC during the last eight (8) weeks of the course
8) require students to use the LMS weekly to learn additional course content; and
9) incorporate active learning activities [1].

## 2 Background

The original research study determined how effective incorporating a MOOC course into a blended learning environment would be on student learning outcomes [5]. This study involved approximately 200 non-STEM students that were taught how to program using Python. In the study, approximately 100 students in a control group learned how to program in a traditional environment while the remaining students (treatment group) learned how to program using a MOOC course in a blended learning environment. A custom textbook was created by an established textbook publisher for both groups. However, the teachers used the textbook to teach programming to the control group. The textbook included chapters on how to program using the Python programming language. The MOOC was used for teaching programming in the experimental group. The learning outcomes in the textbook were aligned with the expected learning outcomes of the MOOC. Both groups had a common midterm exam, several programming laboratory assignments, a common programming assignment, a common rubric and student tutors. The tutors, Computer Science majors, had to learn the MOOC and become familiar with Python [7] [9] [10].

The results of this study produced lessons learned that could add to the success of students learning how to program using a MOOC. Traditionally, in COSC 111 the students are prepared with algorithm and logical thinking exercises before writing a computer program [9] [10]. This best practice was omitted because it was assumed that the MOOC would prepare the students. Also, some of the participants, incoming freshmen, were not familiar with online learning. They had to learn to navigate through the University LMS, MOOC content and how to program. Similarly, the faculty members in the treatment group learned the MOOC content at the same time the students were learning the MOOC content. Furthermore, both groups used different programming development environments which forced the student tutors to learn both programming development environments before offering assistance to the participants [5] [9] [10].

### 2.1 Common Features of Both Research Studies

The course participants, mostly freshmen, were non-STEM majors with no prior computer programming knowledge. The experimental group teachers were required to attend a workshop that introduced the Coursera learning platform. Data collection instruments included both quantitative and qualitative data. Quantitative data collection from the participants consisted of mid-term grades and programming assignment grades. Qualitative data collection consisted of a pre and post surveys for the participants and faculty.

During the first day of class, students were given an overview of the research study and were invited to participate in the study. Participants were offered a small incentive for consenting to participate in the study as well as completing all surveys. The initial survey gathered demographic information as well as gauged the students' familiarity with computing concepts. Initial survey results in both research studies revealed that a majority of the participants had experience using computer devices (e.g. laptops, desktop computers, tablets, mobile devices) and some had online learning experience prior to taking the course. A custom textbook was created by an established textbook publisher for both studies. The MOOC was used for teaching programming in the original study's experimental group and in the extended research study group.

## 3 Instructional Framework

The course redesign of COSC 111 contained two technology driven components: the MOOC content and the LMS [2]. The students accessed the MOOC and LMS from two different websites using two different sets of login credentials. The MOOC was delivered completely online with video enhanced lectures, quizzes, practice exercises and additional programming projects to familiarize the student with the currently taught programming concept and/or topic. CodeSkulptor, was the browser-based programming environment used for the Python language [10][11].

The LMS included the following components for each learning module/unit: an introduction overview, learning objectives, To-do tasks list, reading assignments, activity study guide, supplemental learning materials, interactive media and quizzes [1][3][9]. The quizzes were used to ensure the students accessed the Coursera website and were learning the MOOC content.

## 4 Methodology of the Extended Research Study

The purpose of the extended study was to verify that the best practices discovered from the original research study validate positive changes in a student's ability to learn how to program in Python using a MOOC. This research study utilized one section of COSC 111 with approximately 22 student participants and a faculty member with experience teaching beginning programming to STEM majors. The faculty member teaching this section of COSC 111 was introduced to the MOOC and the Coursera platform prior to offering the course. Mandatory tutoring was available every week and students earned credit for attending tutoring sessions. Additionally, students took weekly quizzes to ensure they learned and retained any new Python concepts covered in the MOOC.

The instructional activities:

Week 1: Students were introduced to searching the Web

Weeks 2 and 3: Students were introduced to representing information digitally and computer operations

Weeks 4 and 5: Students learned about social media.

Week 6: Students implemented a web page using XHTML

Weeks 7 and 8 (two (2) weeks before learning the Python Language): Students were taught a step-by-step problem-solving process that included:

i. writing a basic algorithm without knowledge of the high level language, such as Python
ii. converting the algorithm (English-like statements) into pseudocode (statements written with a combination of English and high-level language)
iii. depicting the pseudocode graphically using a flow chart
iv. using the flow chart to identify variables, constants, assignment statements, and read and write statements
v. defining key programming concepts and terminology

Week 9: Students were given instruction in the following:

i. how to convert all previously completed solved problems described in Weeks 7 and 8 into source code using Python programming language
ii. how to navigate the MOOC (Coursera platform) and all of its contents
iii. how to execute all Python source code written in Python using CodeSkulptor

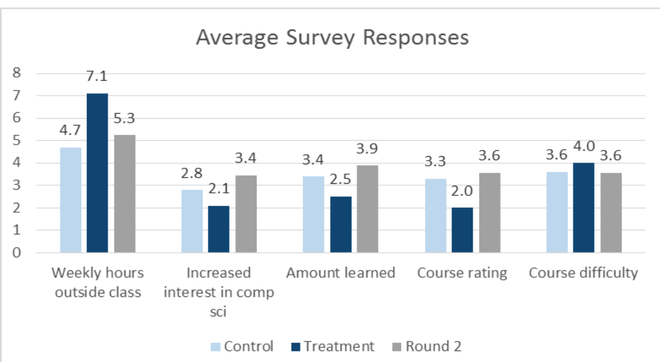
Weeks 10-14: Students created programs using Python programming concepts:

i. variables and assignments
ii. expressions
iii. sequential programming
iv. conditional programming functions
v. lists
vi. interactive applications
vii. button and input fields

# 5 Analysis of the Data

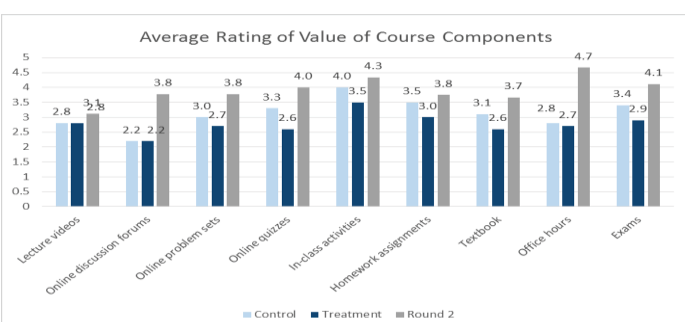
The following graphs show comparison data between the original research study (control and treatment groups) using the MOOC and the extended research study (which included implementation of the best practices). The *Average Survey Responses Table* (Table1), revealed the participants in the extended research study rated the *course difficulty* and the assessed *course rating* similarly. However, the rating *for amount learned* and *increased interest in computer science* were slightly higher than the participants in the original research study.

Table 1: Average Survey Responses



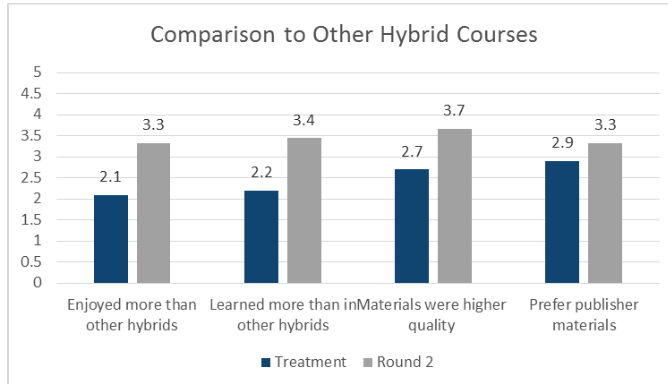

Results in Table 2, *Average Rating of Value of Course Components*, revealed the participants in the extended research study rated the overall value of the course components significantly higher than the participants in the original research groups (control and treatment groups).

Chart 2: Average Rating of Value of Course Components



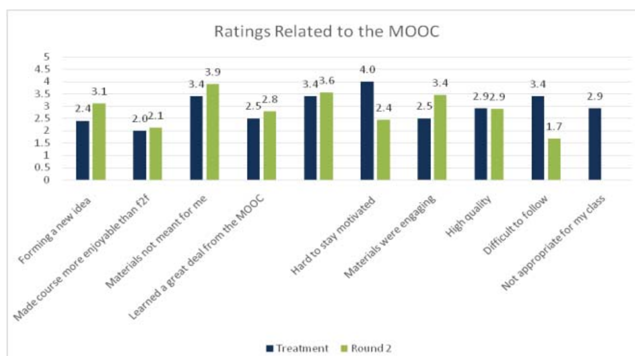

The *Comparison to Other Hybrid Courses Chart* indicates the participants in the extended research study rated this course (extended, redesigned COSC 111) higher than the participants in the original research study's treatment group.

Table 3: Comparison to Other Hybrid Courses




The *Ratings Related to the MOOC Table* (Table 4) revealed the participants in the extended research study thought they could stay motivated because the materials were engaging. The participants did not think that having different instructors (MOOC instructors and the course instructor) was confusing as compared to the rating of the participants in the original treatment group.

Table 4: Ratings Related to the MOOC



# 6 Discussion

The best practices learned from the original research study were successfully integrated in the extended research study. Peer tutoring sessions offered additional assistance with content, concepts, navigation of the MOOC platform, practice programming exercises, compilation and review of the online quizzes offered in the MOOC. Tutoring was mandatory with a weight of 10% of the overall class grade. This grading served as an incentive for students to attend tutoring sessions. Based on Table 2, tutoring had a major effect on the way the participants rated the office hour's category as well as the in-class activities category. Both categories were rated 4.7 and 4.3 respectively out of a 5.0 scale.

A major difference between the original research study and the extended research study is when the MOOC and programming module was introduced to the student. In the original research study, students were introduced the MOOC during the first seven weeks of the semester. In the extended research study, the programming module with the MOOC was introduced during the last eight weeks of the course. Several advantages were uncovered that greatly benefited the students and teacher by offering the programming MOOC during the last eight weeks of the course:

1. The teacher and the student were able to become better acquainted with each other
2. Students are given an ample amount of time to learn the other content of the course (hardware, social networking, networking, web page development) which helped to ease their anxiety of using technology, computers
3. All content covered the first seven weeks is delivered using the LMS
4. The teacher was able to address any technical issues associated with the MOOC prior to using the MOOC in the course

# 7 Conclusion

The extended research study yielded positive results by implementing the best practices for using a MOOC to teach programming to non-STEM students at a Historically Black University. The effectiveness of the study gave insights of the importance of incorporating algorithms and logic exercises prior to writing a computer program. This instructional activity helped students develop the necessary critical thinking skills required to solve a complex computer programming problem. Allowing students time to learn how to navigate through the LMS and the MOOC eased student anxiety, reduced student confusion and frustration as well as decreased the number of onsite technical problems. Providing professional development for faculty to ensure a high level of confidence using technology and understanding how to teach programming decreased the number of faculty problems related to using the MOOC in the instructional environment and encouraged an active learning environment.

# 8 Acknowledgment

This project was supported by grant funding from the Bill and Melinda Gates Foundation. Special acknowledgements to the Ithaka S+R, a strategic consulting and research service, provided by ITHAKA, the University System of Maryland and Dr. J. Warren, Rice University, Computer Science Department.

# 9 References

[1] Bransford, J. D., Brown, A. L., & Cocking, R. R. (Eds.) (1999). How People Learn: Brain, Mind, Experience, and School. Washington, DC: National Academy Press.

[2] Bruff, D.O., Fisher, D.H., McEwen, K.E. & Smith B.E. (2013, June). Wrapping a MOOC: Student perceptions of an Experiment in Blended Learning. Merlot Journal of Online Learning and Teaching. [Online], 9(2), pp. 187-199.Available: http://jolt.merlot.org/vol9no2/bruff_0613.htm

[3] Covington, D. Petherbridge, D. & Warren, S. E. (2005, January). Best Practices: A Triangulated Support Approach in Transitioning Faculty to Online Teaching. Educause Annual Conference. [Online] Available:http://www.westga.edu/~distance/ojdla/spring81/covington81.htm

[4] Goode, J. (2010, March). Connecting K-16 Curriculum & Policy: Making Computer Science Engaging, Accessible, and Hospitable for Underrepresented Students. Presented at SIGCSE '10 Proceedings of the 41st ACM Technical Symposium on Computer Science Education. [Online] Available:http://abraham.cs.uml.edu/~heines/academic/papers/2010sigcse/CD/docs/p22.pdf

[5] Griffiths, R., Chingos, M., Mulhern, C., & Spies, R. "Interactive Online Learning on Campus: Testing MOOCs and Other Platforms in Hybrid Formats in the University System of Maryland. Ithaka S+R., New York, NY, (2014)

[6] Ithaka S+R, "Computing Principles & Technologies: Bowie State University, Side-by-Side Test, Fall 2013" (2014, July). "unpublished"

[7]Jackson, L., Latson, V., & Gross, M. (2011) Redesigning Core Programming Courses Through A Direct Instruction Approach , International Conference Frontiers in Education: Computer Science & Computer Engineering 2011

[8] Keengwe, J. & Kidd, T.T. (2010, June). Towards Best Practices in Online Learning and Teaching in Higher Education. Merlot Journal of Online Learning and Teaching. [Online], 6(2), pp. 533-541.Available: http://jolt.merlot.org/vol6no2/keengwe_0610.htm

[9] Latson, V. (2014) The Impact of MOOC Blended Instruction to Teach Programming. In Learning with MOOCs: A Practitioner's Workshop. Cambridge, MA. 2014.

[10] Latson, V., Jackson, L., Lamar, C., & Stone, D. (2014). The Impact of a Massive Open Online Course (MOOC) to Teach Programming at a Historically Black College and University. American Society for Engineering Education Mid-Atlantic Conference, Swarthmore PA. 2014

# Implementing Elliptic Curve Cryptography

Leonidas Deligiannidis
Wentworth Institute of Technology
Dept. of Computer Science and Networking
Boston, MA 02115, USA
deligiannidisl@wit.edu

***Abstract*—The strength of public key cryptography utilizing Elliptic Curves relies on the difficulty of computing discrete logarithms in a finite field. Diffie-Hellman key exchange algorithm also relies on the same fact. There are two flavors of this algorithm, one using Elliptic Curves and another without using Elliptic Curves. Both flavors of the algorithm rely on the difficulty of computing discrete logarithms in a finite field. Other public key cryptographic algorithms, such as RSA, rely on the difficulty of integer factorization. Both flavors of Diffie-Hellman key exchange algorithm will be discussed in this paper, and we will show implementation details of both of them. Additionally, we will describe what Elliptic Curve Cryptography (ECC) is, and how we can implement different cryptographic algorithms in java, such as digital signatures, encryption / decryption and Key exchange. We will not utilize the java built-in implementations of ECC. Instead, we will use the java programming language as a platform to implement several cryptographic algorithms from the ground up, thus revealing the details of each algorithm and the proofs and reasons these algorithms work. We will describe the theory of ECC and show implementation details that would help students, practitioners, and researchers understand, implement and experiment with such algorithms.**

***Keywords—Elliptic Curve Cryptography, Implementation, Network Security.***

## I. INTRODUCTION

The strength of public key cryptography utilizing Elliptic Curves relies on the difficulty of computing discrete logarithms in a finite field. Diffie-Hellman key exchange algorithm also relies on the same fact. There are two flavors of this algorithm, one using Elliptic Curves [1] and another without using Elliptic Curves [2]. Both flavors of the algorithm rely on the difficulty of computing discrete logarithms in a finite field. Other public key cryptographic algorithms, such as RSA [3], rely on the difficulty of integer factorization. Both flavors of Diffie-Hellman key exchange algorithm will be discussed in this paper, and we will show implementation details of both of them. Additionally, we will describe what Elliptic Curve Cryptography (ECC) is, and how we can implement different cryptographic algorithms in java, such as digital signatures, encryption / decryption and Key exchange. Many [4][5][6][7] implemented these algorithms in Java utilizing java's built-in libraries and they focus more on performance. We will not utilize the java built-in implementations of ECC and we will focus more on functionality instead of performance. Instead, we will use the java programming language as a platform to implement several cryptographic algorithms from the ground up, thus revealing the details of each algorithm and the proofs and reasons these algorithms work. We will describe the theory of ECC and show implementation details that would help students, practitioners, and researchers understand, implement and experiment with such algorithms. We will not evaluate the strengths of each algorithm and the technologies using ECC. A great resource for comparing these algorithms and technologies and discusses their vulnerabilities can be found in [8].

Elliptic Curves (EC) are curves that are also naturally groups. They can be used to form a group that is defined by custom arithmetic operations on its elements. We will first describe these operations geometrically and then algebraically over real numbers. Since ECs will be used to explain cryptographic algorithms, we will focus on ECs over an underlined field $\mathbb{F}_p$ (where $p$ is a prime number); working only with whole numbers. The Elliptic Curve Discrete Logarithm Problem (ECDLP) is the discrete logarithm problem for the group of points on an EC over a finite field. The best known algorithm to solve ECDLP is exponential (or at least sub-exponential since it is faster than exponential in log(p), but slower than polynomial), and that is why EC groups are used for cryptography. In simple terms, the Discrete Logarithm Problem (DLP) is: given an element $n$ in the subgroup generated by a point $g$, find an integer $m$ satisfying $n = g^m$ or else $m=log_g(n)$. If we are working over a large finite field and are given points P and kP, it is very difficult to determine the value of k. This is called the Discrete Logarithm Problem for elliptic curves and is the basis for the cryptographic applications we will see in this paper.

An Elliptic Curve over real numbers consists of the points on the curve, along with a special point O, which is called the point at infinity and is the identity element under the addition operation. The EC can be based on Weierstrass's equation [9] which is of the form $y^2=x^3+ax+b$ where $x$, $y$, $a$ and $b$ are real numbers. The only constraint for the values of $a$ and $b$ is that we do not want the curve to have repeated factors, or else multiple roots; we want the curve to have distinct roots. In other words, the discriminant of $E$, $4a^3+27b^2$ must not equal to zero, which guaranties that the curve is regular and there are no points where the first derivatives of the curve are cancelled out; there are no points with two or more different tangents [10]. The number of Points (or else the order of the curve) on an EC mod p denoted by $\#E(\mathbb{F}_p)$ is given by the Scoof's algorithm [11] and provides the range of the number of points which is:

$$p + 1 - 2\sqrt{p} \leq \#E(\mathbb{F}p) \leq p + 1 + 2\sqrt{p}.$$

Elliptic Curve Cryptography (ECC) [12] has been incorporated in a number of frequently used protocols. ECC is appealing to implementers because it requires smaller key sizes [13] [10] than other public key crypto systems such as RSA [3], while offering the same level of security. This leads to faster and more efficient implementations of algorithms in software and in hardware, and more importantly it enables the design of more energy efficient processors for mobile devices. SSH [14] is a protocol that enables secure remote logins. Key exchange between the server and the clients is implemented using Diffie-Hellman on ECs (ECDH) [1]. Each SSH server has a host key and it is used to authenticate itself to the clients. The clients need to accept and save this key the first time they connect to the server. After the first time, they verify this server host key with the saved host key. The server and the clients, then authenticates themselves by signing a transcript of the key exchange using ECDSA [15] [16] that we will talk about later.

Bitcoin is a distributed peer-to-peer digital crypto currency. It enables users to make online payments directly to other parties without going through a financial institution [17]. To make a payment, a user transfers ownership of bitcoins to another user by attaching his/her ECDSA signature and the public key of the payee at the end of the new transaction.

## II. ECC Operations

### A. *ECC Operations - geometrically*

To add two distinct points P and Q on an EC where $P \neq -Q$, we draw a line through both points. This line intersects the EC in a third point called -R. Reflecting this -R point on the x-axis yields the point R which is R=P+Q, as shown in Figure 1.

If P = -Q, then P+Q= $\mathcal{O}$, as shown in Figure 2. The cubic curves used in ECC are depressed, which means that the square component of the equation is eliminated. The solutions ($r_1$, $r_2$, $r_3$) to such cubic equations is given by $(x-r_1)(x-r_2)(x-r_3)$. Performing the multiplications we get $x^3-(r_1+r_2+r_3)x^2+number$. Because the ECC curve does not have the $x^2$ term, this implies that the sum of its roots is equal to zero. Having the equation of a line y=mx+d, we set the two equations equal to find their intersection points. So, $x^3+px+q = mx+d$ which becomes: $x^3+(p-m)x+(q-d) = 0$ and the solutions are *a*,*b*,*c*. Since the sum of these roots equals zero, *a*+*b* must equal to –*c*. Thus, the reflection of the –*c* is our third point on the curve; and this is the reason we need to reflect the resulting point when we perform additions in ECs.

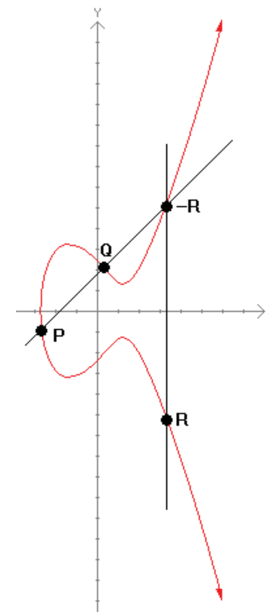



Fig. 1. Adding distinct points P and Q on an Elliptic Curve over ℝ. The result is point R.

Based on addition, we can define multiplication of a point P by a scalar as 2P=P+P, and 3P=2P+P, and 4P=3P+P or 4P = 2P+2P, etc, as shown in Figure 3. Note that when we add a point P to itself, we take the tangent on that point, which intersects the EC on another point, then we reflect that point to get 2P.

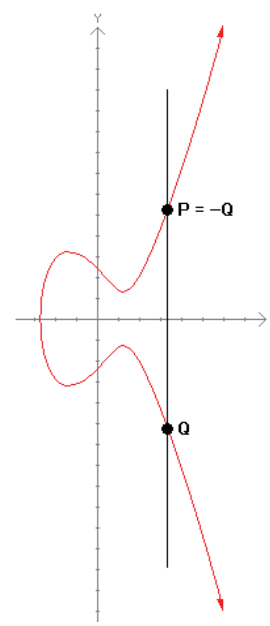



Fig. 2. Adding points P and Q where point P is equal to –Q. The result is the point at infinity $\mathcal{O}$.

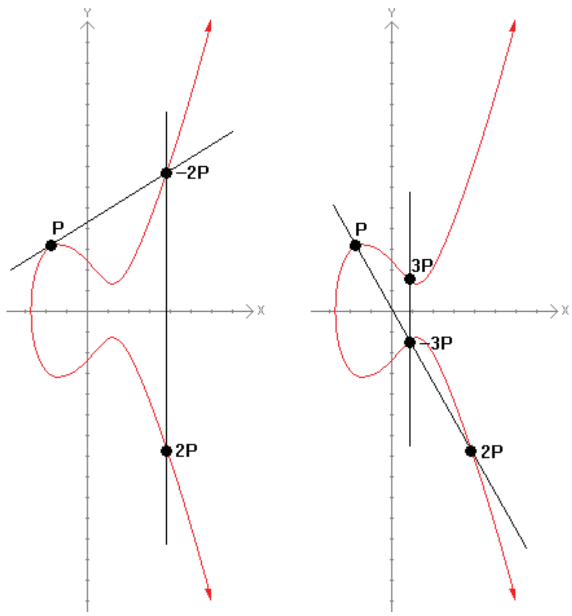



Fig. 3. Doubling the point P produces the point 2P. Adding to that P, produces 3P, etc.

### B. ECC Operations - algebraically

Adding two points $P,Q \in EC$ with coordinates $P=(x_1,y_1)$ and $Q=(x_2,y_2)$, we get point $R=(x_3,y_3)$. Here we have 3 cases to consider [18]:

1. $x_1 \neq x_2$
2. $x_1=x_2$ and $y_1= -y_2$
3. $x_1=x_2$ and $y_1=y_2$

Case 1. The first case involves the addition of two points P and Q that are not negative of each other ($Q \neq -P$). The slope of the line passing through both points is defined as: $L=(y_2-y_1)/(x_2-x_1)$. The coordinates of point $R(x_3,y_3) = P+Q$ are: $x_3 = L^2-x_1-x_2$ and $y_3 =L(x_1-x_3)-y_1$.

Case 2. The second case involves the addition of two points $P=(x_1,y_1)$ and $Q=(x_1,-y_1)$; Q is the reflexion of point P on the x-axis. We can write this as $P + (-P) = \mathcal{O}$, where $\mathcal{O}$ is the special point at infinity (the line passing through P and Q intersects the EC at infinity. Therefore, (x,y) and (x,-y) are inverses with respect to the elliptic curve addition operation. Here we can also see how we can subtract a point $P=(x_1,y_1)$ from point $Q=(x_2,y_2)$ to produce a point R. We simply reverse the sign of the y-coordinate of point P and we perform addition as it is described earlier.

Case 3: The third case involves the addition of a $P=(x_1,y_1)$ with itself: P+P=R; this is also referred to "point doubling". In this case we assume that $y_1 \neq 0$, otherwise we would be looking at case 2 above. In this case, the slope of the line (the tangent here) is calculated as: $L=(3x_1^2+a) / 2y_1$ , where $a$ is the variable $a$ from the equation of the Elliptic Curve. The coordinates of the point $R=(x_3,y_3)$ are calculated the same way as in case 1. Since we are adding a point to itself the calculations can be simplified to: $x_3 = L^2-2x_1$ and $y_3 = L(x_1-x_3)-y_1$.

## III. Elliptic Curves over a Finite Field

To implement cryptographic algorithms we need to work with whole numbers over a finite field, which is an essential property in cryptography; floating point arithmetic is slow and inaccurate due to round off errors. Elliptic Curves over $\mathbb{Z}_p$ can be defined the same way as over $\mathbb{R}$ where p is a prime greater that 3 (in practice p is a large prime number). The equation of the Elliptic Curve now becomes: $y^2 = x^3+ax+b \pmod p$ where $4a^3+27b^2 \pmod p \neq 0$, and $a$, $b$ are in the finite field $\mathbb{F}_p$.

The coordinates $(x_3,y_3)$ of the point R=P+Q are calculated the same way as it is shown in case 1 above, but performed with (mod p): $x_3 = L^2-x_1-x_2 \pmod p$ and $y_3 =L(x_1-x_3)-y_1 \pmod p$. The slope L is calculated as follows: If $P \neq Q$ then $L=(y_2-y_1)(x_2-x_1)^{-1} \pmod p$. If P=Q then the slope L is calculated as follows: $L=(3x_1^2+a)(2y_1)^{-1} \pmod p$, where $a$ is the variable $a$ from the equation of the Elliptic Curve; we can rewrite the calculation of the x-coordinate as: $x_3 = L^2-2x_1 \pmod p$, and $y_3$ stays the same as $y_3 =L(x_1-x_3)-y_1 \pmod p$.

We explained how we can add two distinct points and how to double a point (adding a point to itself). In other words, P+P = 2P. And 2P+P = 3P, 3P+P = 4P, and we can add 3P to 4P to get 7P, etc. We can think of it as the scalar multiplication operation under the additive notation of a point where 7P = P+P+P+P+P+P+P, adding 7 copies of the point P to itself. The strength of Elliptic Curve Cryptography relies on the fact that given points P and Q such that P = kQ is computationally infeasible to find k, and this is the Elliptic Curve Discrete Logarithm Problem (ECDLP).

### A. Properties of Addition on ECs

It can be shown [19] that the curve $E(\mathbb{F}_p)$ under point addition is a commutative / abelian group because of the following properties [20]:

- $P + \mathcal{O} = \mathcal{O} + P = P$ for all $P \in E(\mathbb{F}_p)$, which makes the point $\mathcal{O}$ be the identity under point addition.
- $P + (-P) = \mathcal{O}$ for all $P \in E(\mathbb{F}_p)$, which makes the reflection of a point the inverse of the point.
- $P + (Q+R) = (P+Q) + R$ for all $P,Q,R \in E(\mathbb{F}_p)$, which is the associativity law.
- $P + Q = Q+P$ for all $P,Q \in E(\mathbb{F}_p)$, which is the commutative law.
- $P + Q = R$ for all $P,Q \in E(\mathbb{F}_p)$, $P + Q$ produces result $R \in E(\mathbb{F}_p)$, thus the addition operation is closed on the curve $E(\mathbb{F}_p)$.

The fact that the points on an elliptic curve form an abelian group is behind most of the interesting properties and applications utilizing ECs.

## IV. Java Implementation Details

The algorithms presented here are well known and standardized. We will not present the entire implementation of these algorithms (because of space limitations), instead we will provide implementation hints on how one can implement them. For each algorithm we will provide hints on how some of the important steps of the algorithms can be implemented. Full source code is provided for free upon request.

We do not claim that our implementations outperform any other implementations. We simply provide implementation hints in Java in this paper. The source code can be used as a vehicle to deepen our understanding of the inner workings of such security algorithms.

Java's *BigInteger* object is used almost exclusively for implementing these algorithms. This object contains methods that can operate on large numbers (primes or not) needed for all cryptographic algorithms. Its *probablePrime()* method generates prime numbers of specified number of bits. Other methods allow you to add, subtract, multiply, etc. *BigInteger* objects. One of the important methods that is used frequently is the *modPow()* method which allows you to raise a number to a power and then perform the mod operation, all in this single call. Another method that is used as frequently is the *modInverse()* which allows you to perform a mod operation on the inverse of a number. Based on the *BigInteger* object, we created an *ECC_Point* object. This object allows us to perform additions of Points based on the properties of EC discussed earlier. This object enables us to perform numerical operations considering the point at infinity as well. There are two main functions is this object, one to add two points, and the second to multiply a Point by a scalar. We will be using variable names that begin with **BI** to indicated that the data type of a variable is of *BigInteger*, and **ECP** to indicate that the datatype of a variable is of *ECC_Point*; the complete object is provided in the URL mentioned above.

## V. Diffie-Hellman Key Exchange AlgorithmUsing the Template

The Diffie-Hellman key exchange algorithm is used between two parties to exchange keys securely over a public unsecured communication line. Its strength is based on the fact that while it is easy to calculate exponentials modulo a prime number, it is very hard to calculate discrete logarithms. The Diffie-Hellman Key Exchange algorithm between two parties works as follows:

- Both users agree on a number **q** that is a large prime number.
- Both users also agree on a number **a** that is a primitive root of **q** and **a<q**.
- User A selects a private key **XA** that is **XA<q**.
- User B selects a private key **XB** that is **XB<q**.
- User A calculates her public key $YA = a^{XA} \bmod q$.
- User B calculates her public key $YB = a^{XB} \bmod q$.
- At this point the two users exchange their public keys.
- User A Calculates the Secret key $K=YB^{XA} \bmod q$.
- User B Calculates the Secret key $K=YA^{XB} \bmod q$.

Both users exchanged publicly the **q**, **a**, and their public keys **YA** and **YB**. However, it is very difficult for an eaves dropper to calculate the discrete logarithm and find either **XA** or **XB** knowing **YA**, **YB**, **a**, and **q**. In actuality, the two parties exchanged information in a public domain to securely calculate the shared secret key **K**, which can then be used in other symmetric algorithms.

Implementation hint: To perform the operation $K=YA^{XB} \bmod q$ in Java, we can do:

**BI**_K =**BI**_YA.modPow(**BI**_XB, **BI**_q);

## VI. Diffie-Hellman on Elliptic Curves (ECDH)

The Diffie-Hellman algorithm can also be used over Elliptic Curves between two parties to exchange keys. The algorithm works as follows:

- Both users agree on an elliptic curve *E* over a finite field $\mathbb{F}_q$ such that the discrete logarithm problem is hard in $E(\mathbb{F}_q)$. This implies that both users agree on the Elliptic curve (i.e. $y^2=x^3+ax+b \pmod q$) which includes the values of **a** and **b**, as well as the prime **q**.
- Both users also agree on a base point $G \in E(\mathbb{F}_q)$ so that its order is a large prime.
- **n** is the smallest integer such that **nG=O** (where **O** is the point at infinity).
- User A selects a private key $n_a < n$.
- User A calculates his public key $P_a = n_a G$.
- User B selects a private key $n_b < n$.
- User B calculates his public key $P_b = n_b G$.
- Both users exchange their public keys $P_a$ and $P_b$.
- User A calculates her Secret key $k_a = n_a * P_b$.
- User B calculates her Secret key $k_b = n_b * P_a$.

At this point $k_a = k_b$ because $k_a = n_a * P_b = n_a * (n_b * G) = n_b * (n_a * G) = n_b * P_a = k_b$.

An eavesdropper only sees the curve *E*, the finite field $\mathbb{F}_q$ and the points **G**, $n_a G$ and $n_b G$. If the eavesdropper was capable of solving discrete logarithms in $E(\mathbb{F}_q)$, she could then find $n_a$ from the points **G** and $n_a G$.

Implementation hint: To perform the operation $P_b = n_b G$ in Java, we can do: **ECP**_Pb = **ECP**_G.multiply(**BI**_nb);

## VII. Tripartite Diffie-Hellman

The Diffie-Hellman algorithm on Elliptic Curves can be extended so that three participants are involved to derive a shared key, but we will not discuss this algorithm and its implementation in this paper. This technique requires a single exchange of messages between them [21]. It uses the Weil pairing, but a better approach is to use the Tate pairing, to define the function **F**. Let **E** be the super singular curve $y^2=x^3+1$ over $E_q$ where $q \equiv 2 \pmod 3$. Let **S** be a point on the curve of order **n**. The three users choose a secret number: a, b, c (mod n) respectively. They then calculate their public key as: $P_a$=aS, $P_b$=bS, $P_c$=cS respectively, and they broadcast them. Then each of the three participants computes the shared key as follows, which is the same as $F(S,S)^{abc}$ : User A: $F(P_b, P_c)^a$ , User B: $F(P_a, P_c)^b$ and User C: $F(P_a, P_b)^c$ .

## VIII. ElGamal Digital Signatures

A Digital signature is a mathematical operation on a given data where anyone can verify the entity that signed a message. The verification process should be relatively easy to compute but very hard to forge someone else's signature. If one verifies that the signature is valid, this implies that the entity who claims to have signed the document is the one who actually performed the signage of the document. The following solution relies on the difficulty of computing discrete logarithms over a finite field. Below is the algorithm where User A signs a message and sends it User B. User B then can verify or reject the signature.

- User A selects a generator point $\mathbf{G} \in E(\mathbb{F}_q)$.
- User A selects a secret key **d** (a random number).
- User A computes his public key $\mathbf{PU_A} = \mathbf{dG}$.

Public information: $E, \mathbb{F}_q$, **G**, $\mathbf{PU_A}$.

<u>Message Signature</u>

- User A signs the message **M** by first calculating its hash value: **e**=H(M)
- User A chooses a random number **k** (<u>and keeps it secret</u>) where gcd(k,N) = 1, and $N=\#E(\mathbb{F}_q)$
- User A calculates **P** = **k G**
- User A stores the x-coordinate of point P in **x**
- User A calculates $\mathbf{s} \equiv k^{-1}(e - d\,x) \pmod N$.
- User A transmits the signature **(e, P, s)** followed by the message **M** (User A does not try to keep the message **M** a secret!)

<u>Signature Verification (*needs 3 Point multiplications*)</u>

- User B stores the x-coordinate of point P in **x**
- User B calculates $\mathbf{V_1} = \mathbf{x\ PU_A + s\ P}$
- User B calculates $\mathbf{V_2} = \mathbf{e\ G}$
- User B accepts the signature as valid <u>if and only if $\mathbf{V_1}$ is equal to $\mathbf{V_2}$</u>.

This algorithm works because:

$\mathbf{V_1} = x\,PU_A + s\,P = x\,d\,G + s\,k\,G = x\,d\,G + k^{-1}(e - d\,x)\,k\,G = G\,(\,(d\,x) + e - (d\,x)\,) = e\,G = \mathbf{V_2}$.

<u>Implementation hint</u>: To perform the operation $\mathbf{s} \equiv k^{-1}(e - d\,x) \pmod N$ in Java, we can do:

**BI**_s = **BI**_k.modInverse(**BI**_N).multiply(
**BI**_e.subract(**BI**_d.multiply(**BI**_x))).mod(**BI**_N);

## IX. The Digital Signature Algorithm (DSA) Based on Elliptical Curves (ECDSA)

The Digital Signature Standard [15] is based on the Digital Signature Algorithm [16]. Recently, the ECDSA [22] algorithm emerged that uses Elliptic Curves. ECDSA is similar to ElGamal's signature scheme but it uses a slightly different signature verification method which makes verification of signatures faster [19]. The main difference between ECDSA and ElGamal's digital signature system is that in ElGamal's system the verification process requires three multiplications of an integer times a point, whereas in ECDSA only two multiplications of an integer times a point are required. These multiplications are the most expensive parts of these algorithms.

- User_A wants to sign a message **m**, which is an integer or most likely the hash of the document to be signed.
- User_A chooses an elliptic curve $E$ over a finite field $\mathbb{F}_q$ where q is a prime number
- User_A chooses a base point $\boldsymbol{G} \in E(\mathbb{F}_q)$ of order $\boldsymbol{n}$; n is the smallest positive integer that nG= O, which is also the number of points on the curve.

<u>Key Generation</u>

- User_A select a random $\mathbf{d} \in [1...n-1]$.
- User_A computes $PU_a = dG$, which is a point on the curve. **d** is the private Key and $\mathbf{PU_a}$ is the public key of User_A.

Public Information: $E, \mathbb{F}_q, n, G, PU_a$.

<u>Signature Generation of message m</u>

- User_A select a random number **k**, $(1 \le k < n)$
- User_A computes P=(**x**,y)=kG and r = **x** mod n. If r is zero restart
- User_A computes e = H(m) H is SHA-1 (160 bit hash)
- User_A computes $s \equiv k^{-1}(e+dr) \bmod n$. If s is zero restart

The Signature is the pair **(r,s)**

<u>Signature Verification</u>

- User_B verifies that r and s are in [1...n-1]
- User_B computes e=H(m)
- User_B computes $w \equiv s^{-1} \bmod n$
- User_B computes $u_1 = ew$ and $u_2 = rw$
- User_B computes $X=(\mathbf{x},y) = u_1G+u_2PU_a$
- If X is the infinity point, User_B rejects the signature, else she computes: v = **x** mod n

User_B accepts the signature iff v is equal to r, since

r = **x** mod n from P=(**x**,y), that User A computes, and

v = **x** mod n from X=(**x**,y), that User B computes.

This algorithm works because:

$\mathbf{X} = u_1G+u_2PU_a = ewG+rwPU_a = es^{-1}G+rs^{-1}PU_a = es^{-1}G+rs^{-1}dG = s^{-1}(e+rd)G = kG = \mathbf{P}$

<u>Implementation hint</u>: To perform the operation $s \equiv k^{-1}(e+dr)$ mod n in Java, we can do:

**BI**_s = **BI**_k.modInverse(**BI**_N).multiply(
**BI**_e.add(**BI**_d.multiply(**BI**_x))).mod(**BI**_N);

## X. ElGamal Public Key Encryption

Public Key Encryption is the mathematical operation to a message that enables only the recipient to decrypt the message. There are two families of encryption algorithms: symmetric and asymmetric. In symmetric algorithms, the same key that encrypts can also decrypt a message. In asymmetric encryption, or else public key encryption, one key encrypts and the other decrypts – one key is kept secret and the other is made public. Generally, public key crypto-systems are slower than symmetric crypto-systems. Therefore, it is common to use public key systems, such Diffie-Hellman, to negotiate and establish a key that is then used in a symmetric crypto-system.

Here is the ElGamal's encryption algorithm where User A encrypts a message and sends it to User B. Then only User B is able to decrypt the recover the original message.

<u>Initialization</u>

- User B chooses a generator point $\mathbf{G} \in E(\mathbb{F}_q)$.

- User B chooses a secret integer **d.** (**d** is user's B private key).
- User B computes $\mathbf{PU_B}$ = dG. ($\mathbf{PU_B}$ is user's B public key).

Public Information: $E$, $\mathbb{F}_q$, G, $PU_B$

Encryption

- User A expresses her message M $\in E(\mathbb{F}_q)$.
- User A chooses a secret session random integer **k**. It is important that a different k is used each time.
- User A computes $\mathbf{M_1}$ = kG
- User A computes $\mathbf{M_2}$ = M + $kPU_B$
- User A sends to User B: ($M_1$, $M_2$)

Decryption

- User B computes and recovers M = $M_2$-$dM_1$

This algorithm works because:

$M_2-dM_1 = (M+kPU_B) - d(kG) = M + kdG - kdG = M$

Implementation hint: To perform the operation M=$M_2$-$dM_1$ in Java, we can do:

```
ECP_M = ECP_M2.subract( ECP_M1.multiply(
BI_d)).mod(BI_q);
```

## XI. MASSEY-OMURA ENCRYPTION

In Massey-Omura encryption scheme, both users agree on an elliptic curve $E$ over a finite field $\mathbb{F}_q$ such that the discrete log problem is hard in $E(\mathbb{F}_q)$. This algorithm works as follows. User A places a lock on the message M and sends it to User B. User B places another lock on the message and sends it back to User A. User A then removes his lock (leaving the message with only User B's lock on, and sends it back to User B. User B then removes his lock to retrieve the message that User A sent him. Below is the algorithm. It is important to notice here that "removing a lock" requires the multiplication of a point with one's inverse private key. To do this, both users need to know **N**=#$E(\mathbb{F}_q)$. If **d** is the private key, the **d'** is the inverse of **d** in $\mathbb{F}_q$ which **d*d'** = 1.

- User A wants to encrypt a message M ∈ $E(\mathbb{F}_q)$.
- User A chooses a secret integer $\mathbf{d_a}$ with gcd($d_a$,N) = 1.
- User A computes $\mathbf{M_1}$ = $d_aM$ and sends it to User B.
- User B chooses a secret integer $\mathbf{d_b}$, with gcd($d_b$,N) = 1.
- User B computes $\mathbf{M_2}$ = $d_bM_1$ and sends it back to User A.
- User A computes $d_a^{-1} \in \mathbb{Z}_n$
- User A calculates $\mathbf{M_3}$ = $d_a^{-1}$ $M_2$ and sends it to User B.
- User B computes $\mathbf{d_b^{-1}} \in \mathbb{Z}_n$
- User B calculates $\mathbf{M_4}$ = $d_b^{-1}$ $M_3$.

$M_4$ = M because: $M_4 = d_b^{-1} M_3 = d_b^{-1} d_a^{-1} M_2 = d_b^{-1} d_a^{-1} d_bM_1 = d_b^{-1} d_a^{-1} d_b d_aM = M$

Implementation hint: To find the inverse of a number in $\mathbb{F}_q$ of order N, we perform the following (finding $d_a^{-1}$ knowing $d_a$):

```
BI_daInv = BI_da.modInverse(BI_N);
```

## XII. ELLIPTIC CURVE INTEGRATED ENCRYPTION SCHEME (ECIES)

ECIES is another encryption scheme. ECIES is the most extended encryption scheme utilizing ECs. It is defined in ANSI X9.63 [23], IEEE 1363a [24], ISO/IEC 18033-2 [25] and SECG SEC 1 [26]. In [27] the authors compare the different ECIES versions that are implemented by different security organizations and standards. Both users agree on an elliptic curve $E$ over a finite field $\mathbb{F}_q$ so that the discrete log problem is hard for $E(\mathbb{F}_q)$. Both users also agree on a point **G** ∈ $E(\mathbb{F}_q)$. **N**=#$E(\mathbb{F}_q)$ is the number of points on the curve; the order of the curve. We need 2 different hash functions $H_1$ and $H_2$ and a symmetric encryption algorithm. An advantage of ECIES over ElGamal's and the Massey-Omura public key encryption methods is that the message M is not represented as a point on the curve, on the contrast it is any sequence of bytes. Here User B wants to send an encrypted message M to user A.

Initialization

- User A chooses a secret integer **d**. (this is the secret key).
- User A computes $PU_A$ = dG. (this is the public key).

Public Information: ($E$, $\mathbb{F}_q$, N, G, $PU_A$)

Encryption

- User B chooses a random number **k** where: $1 \le k \le N - 1$
- User B computes **R** = kG
- User B computes **Z** = $kPU_A$
- User B computes the hash of R and Z: $\mathbf{R_h}$=$H_1$(R) and $\mathbf{Z_h}$=$H_1$(Z).
- User B encrypts message M using a symmetric algorithm and the key $R_h$ : **C**=$E_{Rh}$(M)
- User B computes the hash of C and $Z_h$: $\mathbf{t_1}$=$H_2$(C) and $\mathbf{t_2}$=$H_2$($Z_h$)
- User B sends to User A: (R, C, $t_1$, $t_2$)

Decryption

- User A computes **Z** = dR
- User A computes $\mathbf{R_h}'$=$H_1$(R) and $\mathbf{Z_h}'$=$H_1$(Z)
- User A computes $\mathbf{t_1}'$=$H_2$(C) and $\mathbf{t_2}'$=$H_2$($\mathbf{Z_h}'$)
- If $t_1$ is not equal to $t_1'$ or $t_2$ is not equal to $t_2'$, reject the cipher-text C and don't even try to decrypt it, else continue.
- User A computes M = $D_{Rh'}$(C). (Note here that $R_h' = R_h$).

For this algorithm to work, the Z values computed by both users must be the same, and they are because:

(User A computes): Z = dR = dkG

(User B computes): Z = $kPU_A$ = kdG=dkG

Implementation hint: To compute the Z point, user B can do the following: **ECP_Z** = **ECP**_$PU_A$.multiply(k);

## CONCLUSION

We presented the underlying number theory of Elliptic Curve Cryptography in a simple to understand way. We explained several algorithms and the mathematical reasons why they work. We also explained how these algorithms can be implemented using the java language and platform. We provide the complete source code of these implementations for researchers and students to experiment with and discover the beauty and elegancy of Elliptic Curve Cryptography.

## REFERENCES


[1] U.S. Department of Commerce/National Institute of Standards and Technology (NIST). Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography. Special Publication 800-56A Revision 2, May 13 2013. http://csrc.nist.gov/publications/nistpubs/800-56A/SP800-56A_Revision1_Mar08-2007.pdf (retrieved Feb. 2015).

[2] Whitfield Diffie and Martin E. Hellman. "New Directions in Cryptography". IEEE Transactions on Information Theory Vol. 22 (6). p644–654, Nov. 1976.

[3] R. L. Rivest, A. Shamir, and L. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". Communications of the ACM, Vol.21, Issue 2, pp120-126, Feb. 1978.

[4] Andrew Burnett, Keith Winters, and Tom Dowling, "A Java implementation of an elliptic curve Cryptosystem". Proceedings of the inaugural conference on the Principles and Practice of programming, 2002 and Proceedings of the second workshop on Intermediate representation engineering for virtual machines, (PPPJ '02/IRE '02), 2002 Pages 83 - 88 National University of Ireland Maynooth, County Kildare, Ireland, Ireland ISBN: 0901519871

[5] V. Gayoso Martınez, L. Hernandez Encinas, and C. Sanchez Avila. "A Java Implementation of the Elliptic Curve Integrated Encryption Scheme." The 2010 International Conference on Security and Management (SAM'10), 2010

[6] Johann Großschadl, Dan Page, and Stefan Tillich. "Efficient Java Implementation of Elliptic Curve Cryptography for J2ME-Enabled Mobile Devices." Information Security Theory and Practice. Security, Privacy and Trust in Computing Systems and Ambient Intelligent Ecosystems Lecture Notes in Computer Science Volume 7322, 2012, pp 189-207

[7] V. Gayoso Martınez and L. Hernandez Encinas. "Implementing ECC with Java Standard Edition 7". International Journal of Computer Science and Artificial Intelligence Vol. 3 Iss. 4 pp134-142, Dec 2013

[8] Joppe W. Bos, J. Alex Halderman, Nadia Heninger, Jonathan Moore, Michael Naehrig, and Eric Wustrow. "Elliptic Curve Cryptography in Practice" In Proc. Of the 18th conf. on Financial Cryptography and Data Security (FC 2014) Christ Church Barbados. March 3-7, 2014. Springer Nov. 8 2014.

[9] Joseph. H. Silverman. "The Arithmetic of Elliptic Curves (2nd ed.), Springer-Verlag, New York, ISBN: 978-0-387-09493-9, 2009.

[10] Hankerson, Darrel, Menezes, Alfred J., Vanstone, Scott. "Guide to Elliptic Curve Cryptography". Springer-Verlag, New York, NY, ISBN: 978-0-387-95273-4, USA, 2004.

[11] René Schoof. "Elliptic Curves over Finite Fields and the Computation of Square Roots mod p". Mathematics of Computation 44, (1985), p483-494.

[12] Victor S. Miller. "Use of Elliptic Curves in Cryptography". Advances in Cryptology – CRYPTO 1985 Proceedings. Lecture Notes in Computer Science Vol. 218 pp417-426, Springer 1986.

[13] I. F. Blake, G. Seroussi, and N. P. Smart. "Elliptic Curves in Cryptography". London Mathematical Society Lecture Note Series Vol. 265. Cambridge University Press, Cambridge, 1999 ISBN 0521-653746. Reprinted in 2000.

[14] D. Stebila and J. Green. "Elliptic Curve Algorithm Integration in the Secure Shell Transport Layer". RFC 5656, Dec. 2009. http://tools.ietf.org/html/rfc5656 Retrieved Feb. 2015.

[15] Digital Signature Standard (DSS), Federal Information Processing Standards Publications FIPS 186-4 July 2013 National Institute of Standards and Technology (NIST) doi:10.6028/NIST.FIPS.186-4 U.S. Department of Commerce http://csrc.nist.gov/publications/PubsFIPS.html , http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf. Retrieved Feb. 2015.

[16] Digital Signature Algorithm US patent 5231668A Jul. 27 1993 Inventor David W. Kravitz. http://www.google.com/patents/US5231668 Retrieved Feb. 2015.

[17] Satoshi Nakamoto. "Bitcoin: A Peer-to-Peer Electronic Cash System. Oct. 2008 https://bitcoin.org/bitcoin.pdf, Retrieved Feb 2015.

[18] Douglas R. Stinson "Cryptography Theory and Practice 3rd Edition Discrete Mathematics and its applications". Series editor Kenneth H. Rosen. Publisher: Chapman & Hall/CRC Taylor & Francis Group Boca Raton London New York, ISBN13: 978-1-58488-508-5, 2006.

[19] "ELLIPTIC CURVES" NUMBER THEORYAND CRYPTOGRAPHY SECOND EDITION, Discrete Mathematics and its Applications" Series Editor Kenneth H. Rosen 2008 by Taylor & Francis Group, LLC. Chapman & Hall/CRC. Taylor & Francis Group 6000 Broken Sound Parkway NW, Suite 300. Boca Raton, FL 33487-2742. Chapman & Hall/CRC is an imprint of Taylor & Francis Group, an Informa business. ISBN: 13: 978-1-4200-7146-7

[20] William Stallings, "Cryptography and Network Security: Principles and Practice" fourth edition. ISBN: 0-13-187316-4 Pearson / Prentice Hall Upper Saddle River, NJ, 2006

[21] Antoine, Joux. "A one round protocol for tripartite Diffie-Hellman". In Algorithmic Number Theory, Ed. W. Bosma, Vol. 1838 of Lecture Notes in Computer. Sci., pp 385–393. Springer- Verlag, Berlin, 2000.

[22] The FIPS 186-4 Elliptic Curve Digital Signature Algorithm Validation System (ECDSA2VS) Updated March 18 2014. National Institute of Standards and Technology (NIST). Information Technology Laboratory. http://csrc.nist.gov/groups/STM/cavp/documents/dss2/ecdsa2vs.pdf Retrieved Feb. 2015.

[23] American National Standards Institute. "Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography". ANSI X9.63, 2001.

[24] Institute of Electrical and Electronics Engineers. Standard Specifications for Public Key Cryptography - Amendment 1: Additional Techniques. IEEE 1363a, 2004.

[25] International Organization for Standardization / International Electrotechnical Commission. Information Technology - Security Techniques - Encryption Algorithms - Part 2: Asymmetric Ciphers. ISO/IEC 18033-2, 2006.

[26] Standards for Efficient Cryptography Group. Recommended Elliptic Curve Domain Parameters. SECG SEC 1 version 2.0, 2009.

[27] V. Gayoso Martínez, F. Hernández Álvarez, L. Hernández Encinas. "Analysis of ECIES and Other Cryptosystems Based on Elliptic Curves". Journal of Information Assurance and Security 6(4) : 285-293 (2011).

# From Big Data to Smart Data

## Teaching Data Mining and Visualization

**Antonio Sanchez, Lisa Burnell Ball**
Department of Computer Science
Texas Christian University, Fort Worth, Texas, USA
{a.sanchez-aguillar,l.ball}@tcu.edu

**Abstract -** *The most important part of big data processing is to create knowledge by converting data into smart data. Smart data necessitates both data mining and visualization techniques. We believe it is important to cover these concepts in an undergraduate computer science curriculum and thus have been teaching a data mining and visualization course for several years. In visualization we concentrate on representation and the interaction with data. For data mining we cover several topics, including entropy and Information Gain to select attributes best for classification and prediction. For processing text, statistical algorithms based on Bayes' Theorem are used for filtering and tokenization of the data. Supervised and unsupervised attribute selection is addressed in part by distinguishing between classification and association. The discussion of the kappa metric and confusion tables using cross validation is also covered, as is the use of instance-based learning based on neighborhood correlation for classification and clustering. Finally, for dealing with very large datasets the use of MapReduce and related models are taught.*

**Keywords**: smart data, data mining, data visualization, undergraduate courses component

# 1 Introduction: From Information to Knowledge

*"Knowledge is experience, the rest is information"*
Albert Einstein

Data mining and visualization are essential considering the exabytes of data now available. To go from Big Data to Smart Data, we need to think in terms of Artificial Intelligence (AI) machine learning algorithms to perform the analysis, selection, and definition of patterns, i.e. creating knowledge rather than processing information using traditional database applications. According to McKinsey [9] smart data analytics will be the key to competition, productivity and innovation. Regardless of the domain (e.g. health care, public service, manufacturing), a large shortage of experts is predicted by 2018 with more than 150,000 unfilled openings for those with the required analytical skills [1]. It is time to start doing something and to rigorously teach these skills to undergraduate Computer Science (CS) and Information Technology (IT) students. We have taken this approach and in this paper we discuss the topics we consider relevant to teach including the acquisition of analytical skills using AI machine learning topics.

It is clear that there must be more emphasis on knowledge and choice than on information. Patterns, clusters and classifications are at best the truth but in many cases we only have probable meaningful intelligence and have to make an adequate selection. Data driven analysis is a process by which machine learning algorithms are able to obtain such adequate solutions. It must be emphasized that there is not a single data mining approach, but rather a cadre of techniques that work alone or in combination with each other.

An important element in smart data deals with the proper visualization of the data; in order to obtain the desired response from an audience data must be beautiful, coherent and interactive. Otherwise we lose the audience regardless of the relevance of the data.

The heart of our pedagogical approach assumes a good deal of computer programming, yet we use powerful programming Java libraries, specifically Processing [11] for Data Visualization, Weka [6] for Data Mining and Hadoop [12] for Map/Reduce stream processing. CS and IT students must be aware that discrete programming is the heart of any software development and a desire for it is what makes us love our profession.

# 2 From Table Data to Data Visualization

*"The greatest value of a picture is when it forces us to notice what we never expect to see."*
John W. Tukey

We begin our course teaching data visualization, rather than going from the data towards the user. We believe that students must be made aware that any table or graph must have a natural narrative to engage the audience. Understanding the questions that need to be answer is the way to approach a visualization project. After this then go and look for the necessary data. Visualization requires indeed some data mining, but mostly the requirements deal with acquiring, parsing and filtering the data is where emphasis should be placed. Creating meaningful interactive representations is the key to a successful visualization project.

On the subject of representation and interaction we must be careful to direct our interest into representing the dependent variables not in various dimensions; there are only three that we can visualize. Rather try to use in a two dimensional scheme using other means to represent other dimensions. Interaction is to be used to allow the user to dynamically modify the independent variables thus producing the necessary enticement to capture the audiences. Rather than plotting a simple $Y = F(X)$ as a graph or a table of values let the user vary the values of $X$ and let $Y$ change dynamically, as shown in the example presented below.

The example we present is the result of a project assigned to a student [13] and is shown in Fig. 1. In his case rather than presenting tables with public county data for the number of marriages in Texas, he implemented a solution using a SVG map, adding color to the counties with higher values (the dependent variable). The interaction is achieved by modifying the independent variables for month and year thus providing a engaging visualization where patterns can be perceived easily.

For the course we use Ben Fry's processing [3] but not necessarily as a language but rather as a set of Java jar libraries that provide the flexibility to add more jar libraries to enhance any programming project. The benefit of this approach is to have access to the large processing API that reduces the gory details of visual programming as well as file processing and interaction. Moreover, many other capabilities such as data mining implementations are available. Well versed in programming, students can use Eclipse or any other IDE to complete their assignments.

# 3 Divide and Conquer using Entropy

In Cybernetics entropy has been used as a measurement of diversity and information gain. In his seminal work, W. Ross Quillian approached data mining using entropy to obtain classification trees. To this day C4.5 and C5 are a good approach to do so. The drawback lies in the requirement of preprocessing to create an a priori model. In our course we use Weka [7] libraries to create classification trees; again by using the Weka libraries with Java, students can add more features to their projects, such as nice visualizations as described before. Most importantly, entropy can be used in another crucial task in data mining: attribute classification. This is in the selection of the best feature attributes to be used in the classification, while disregarding those that do not provide useful information gain, reducing the computation time. Entropy as simply defined as $\sum -pLog_2p$ is a natural and efficient way to do it. Weka has two classes (InfoGain, GainRatio) that use this metric to evaluate the features or attributes, along with the ranker class for the search method that allows fewer attributes in datasets [8]. Yet the a priori pruning suggested here should be used with care and expertise; this is to say that you must know your data. But then again we are in the realm of knowledge processing and so choices must be made. Generally speaking in a denormalized table the use of entropy can reduce the number of attributes to be considered to a manageable number of less than 10.

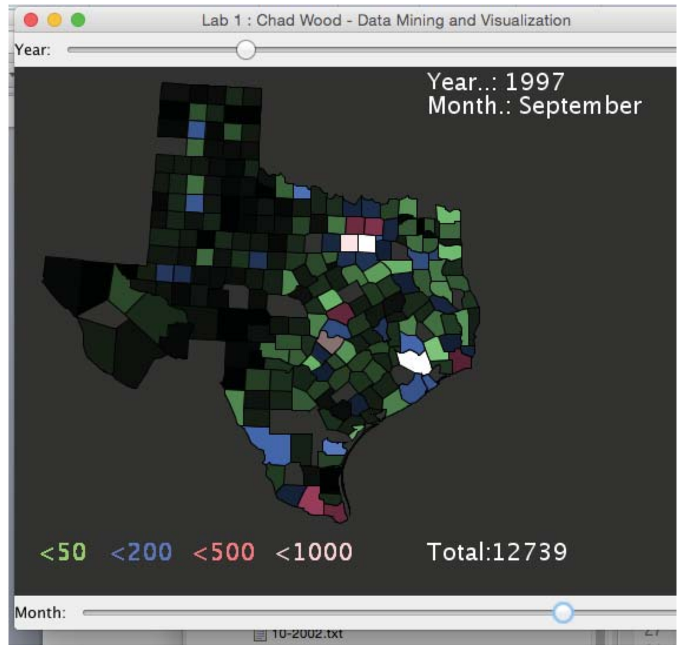



Fig. 1. Example student visualization project to represent marriage rate by county [13].

# 4 Structured Data to Text Classification

Unstructured text processing is an ever increasing application. For example, search engines rely on it. The traditional SQL approach in relational databases prescribes structured, normalized tables. NoSQL alternatives are gaining popularity as alternatives to the relational model. In any case preprocessing text before mining it is an important application to be studied. String tokenization, Stemming and the use of stopword dictionaries are necessary requirements when dealing with text. The computation of text frequencies using IDTF or TFT transforms along with normalization of word frequencies is a detailed and delicate task in the field. Fortunately Weka provides a set of filters that helps in these computations. An example by Hall [6] is shown in Fig. 2.

Classification in this case is done using Bayes' Rule of conditional probability simply defined as

$$p(H|E)=p(E|H)p(H)/p(E) \quad (1)$$

Although the number of attributes or feature words may be large, the Naïve Bayes approach, which assumes independence among features, is common. Along with this a modified Multinomial approach is also common practice. In any case a Laplace Estimator is used to avoid zero frequencies. We use this approach in the classroom to train text messages for rejection or acceptance.

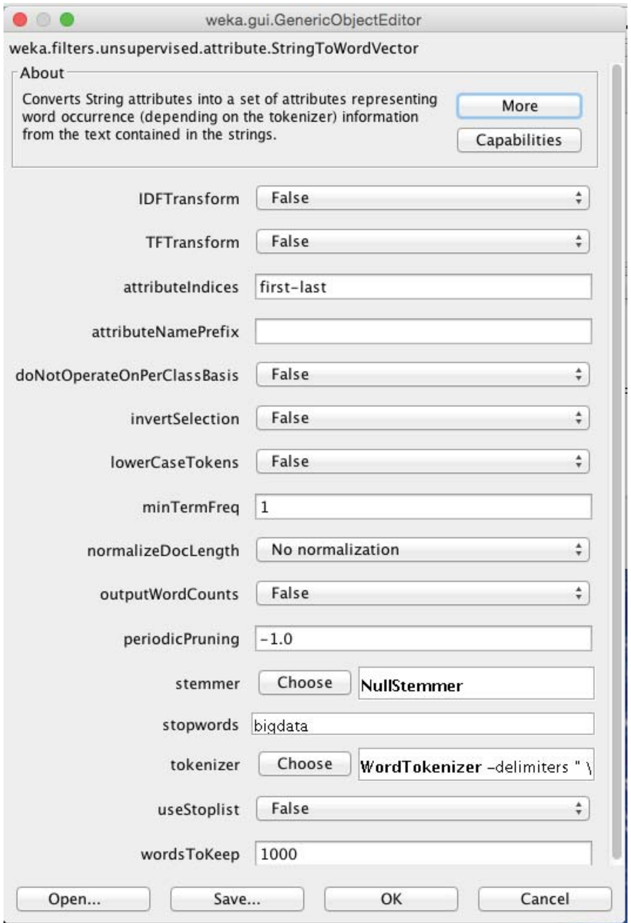



Fig. 2. Weka GUI interface showing a filter for string tokenization.

## 5 Knowledge Validation

The fact that data mining is a selection among alternatives, an important aspect in data mining is that of knowledge validation. Fortunately truth as defined by Saint Thomas is pretty much a metric and not a fixed value, allowing us to discuss in the classroom how good our data mining is. When doing classification using various machine learning algorithms we end up with a single measurement: Success = Right Classifications/Total number of classifications. Provided that we use the normal distribution and the law of large numbers, we can determine confidence level for a classification by solving the probability *p* as suggested by Hall et al. [7]. The important point in this solution is that as the number of cases increase the interval range can be reduced, thus obtaining more precise estimates. Indeed the size of the dataset is ever more important as stated by Halevy, Norvig, and Pereira [5] and Domingos [2]. Added to this a ten-fold cross validation is also a good way to validate a classification. The use of confusion matrices and a Kappa measurement is a standard way to see the performance of a machine learning algorithm since we are reducing the results of a random classificator.

## 6 Supervised and Unsupervised Classification

Affinity association versus classification is yet another important topic of discussion in class. Mostly because unsupervised attribute selection is the standard approach when confronted with relationships among various activities seen as attributes. The so-called “market basket analysis” occurs when retailers seek to understand the purchase behavior of customers. Weka provides both filtered and unfiltered a priori associators that students can use for Affinity Analysis in contrast to the Prism used in traditional supervised classification.

## 7 Correlation: the Basis of Big Data Mining

Finally we discuss the most commonly used approach to data mining, instance-based machine learning or memory-based learning based on correlation. In this case instead of performing explicit a priori models, new instances are compared with previous neighborhood instances seen in training, which have been stored in memory. This is the reason for calling this approach lazy learning; in reality it should be called delayed learning. The hypotheses are created dynamically thus allowing data and complexity to grow. Its common practice may be attributed to its ability to adapt to previously unseen data. Instances are compared by a simple numerical distance measurement such as the Euclidean(squared values) or Manhattan (absolute values) using the following simple metric considering all the attributes *(attribute 1 to k)* or features for two instances in question *(i1, i2):*

$$D = SQRT\ [\ (a_1^{i1}-a_1^{i2})^2 + (a_2^{i1}-a_2^{i2})^2 + \ldots + (a_k^{i1}-a_k^{i2})^2\ ] \quad (2)$$

Using the approach, attribute normalization is required as well as the assignment of maximum values. Examples of this type of classification are the k-nearest neighbor algorithm and kernel machines for limited search. As more data is added memory management becomes an issue when storing all training instances; care should be taken to avoid over fitting to noise in the training set. Clustering is a widely used unsupervised learning approach in which instances are grouped according to a (1) natural center of mass using a distance based measurement or K means, (2) a probabilistic expected maximization, or (3) self organizing. Students need to know how and when to employ clustering. Weka provides useful classes for clustering, examples of which are shown in Fig. 3 [7] and Fig 4 [7].

## 8 Hadoop

A course in Big Data would not be complete without the MAP/REDUCE stream programming model for processing and generating large datasets using a parallel distributed architecture. With data mining an initial set of MAP/REDUCE steps can be perceived as the filtering required to prepare the dataset providing redundant storage.

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances          19
Incorrectly Classified Instances         5
Kappa statistic                          0.6262
Mean absolute error                      0.2262
Root mean squared error                  0.3165
Relative absolute error                 59.8856 %
Root relative squared error             72.4707 %
Total Number of Instances               24

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall
                 0.8      0.053      0.8        0.8
                 0.75     0.1        0.6        0.75
                 0.8      0.222      0.857      0.8
Weighted Avg.    0.792    0.167      0.802      0.792

=== Confusion Matrix ===

  a  b  c   <-- classified as
  4  0  1 |  a = soft
  0  3  1 |  b = hard
  1  2 12 |  c = none
```

Fig. 3. Weka output for instance based classification using a simple dataset for contact lenses.

```
kMeans
======

Number of iterations: 2
Within cluster sum of squared errors: 47.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                   Cluster#
Attribute            Full Data          0          1
                        (24)         (12)       (12)
====================================================
age                    young        young      young
spectacle-prescrip     myope        myope      myope
astigmatism               no           no         no
tear-prod-rate       reduced       normal    reduced
contact-lenses          none         soft       none
```

```
                 Cluster
Attribute            0     1     2
                  (0.58)(0.25)(0.17)
==============================
age
  young              5     3     3
  pre-presbyopic     5     4     2
  presbyopic         7     2     2
  [total]           17     9     7
spectacle-prescrip
  myope              8     3     4
  hypermetrope       8     5     2
  [total]           16     8     6
astigmatism
  no                 8     6     1
  yes                8     2     5
  [total]           16     8     6
tear-prod-rate
  reduced           13     1     1
  normal             3     7     5
  [total]           16     8     6
contact-lenses
  soft               1     6     1
  hard               1     1     5
  none              15     2     1
  [total]           17     9     7
```

Fig. 4. Weka output for K means and EM clustering with the data used in Fig. 3.

After this first phase data mining, algorithms can be used. The standard software for this approach is Hadoop [12]. For clustering and classification we use mahout [4]. The benefit of using these as Java libraries is that we can combine our programs with other libraries as discussed before. When confronted with massive databases traditional similarity instance-based learning may be substituted with co-occurrence matrix computations that can be obtained using MAP/REDUCE cycles. This is the approach suggested by Owen, Anil & Dunning [10]. Using mahout clustering can also be represented in terms of MAP/REDUCE cycles. Note that Hadoop/mahout can be run both in truly distributed architectures or a single machine thus allowing our Java students to apply their data mining programming skills on their own.

# 9 Conclusions

*"An innovation is a transformation of practice in a community. It is not the same as the invention of a new idea or object. The real work of innovation is in the transformation of practice."*
Peter Denning

Indeed research in Big Data and data mining is a hot topic today, however if these breakthroughs are not taught at the undergraduate level the topic does not become an innovation. Just as Peter Denning states, innovation requires new routines to be created and so teaching these concepts becomes an important part of the innovation process. At our institution we teach a course on data mining and visualization open to juniors and seniors; mainly we are interested in savvy programmers that have been exposed to both data structures and database concepts. The course exposes students to current trends, problems, and technologies. Team projects, active learning, and small class size have all contributed to the success of the course. In formal student assessment, students overwhelmingly report the course as challenging and important. We encourage departments to include a course in data mining and visualization in their CS and IT undergraduate programs.

# 10 References

[1] T. H. Davenport and D. J. Patil, "Data Scientist: The Sexiest Job of the 21st Century," Harvard Business Review, pp 70-76, October 2012.

[2] P. Domingos, "A Few Useful Things to Know About Machine Learning," Communications of ACM , vol 55(10), pp 78-87, 2012.

[3] B. Fry, Visualizing Data: Exploring and Explaining Data with the Processing. CA: O'Reilly Media, 2007.

[4] P. Giacomelli. Apache Mahout Cookbook, UK: Packt Publishing Ltd., 2013.

[5] A. Halevy P. Norvig, and F. Pereira, "The Unreasonable Effectiveness of Data", IEEE Intelligent Systems, pp. 8-12, March/April 2009.

[6] M. Hall, F. E. Holmes, G. Pfahringer, B. Reutemann, and I. Witten, "The Weka Data Mining Software: An Update", SIGKDD Explorations, vol 11(1), 2009.

[7] M. Hall, I. Witten, and E. Frank, "Data Mining: Practical Machine Learning Tools and Techniques 3rd ed". MA: Morgan Kaufmann, 2011.

[8] B. Kaluza, Weka How-to. UK: Packt Publishing Ltd, 2013.

[9] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, and H. Byers, Big data: The next frontier for innovation, competition, and productivity. NY: McKinsey Global Institute, 2011.

[10] S. Owen, R. Anil, and T. Dunning. Mahout in Action. NY: Manning Publications Co., 2011.

[11] Processing Foundation, *Processing* http://processing.org Accessed March 2, 2015.

[12] T. White, Hadoop: The Definitive Guide 3rd ed. CA: O'Reilly Media, 2011.

[13] C. Wood, "Lab 1 Presentation in CoSc 40023 Data Mining & Visualization", Fort Worth: TCU, 2014, unpublished.

# Toward Smart Content in Adaptive Learning Systems: Potential, Challenges, and Solutions

[1]Ketly Jean-Pierre, [1]Marlon Mejias, [1]Dr. Legand Burge III Ph.D., [2]Dr. Kyndra Middleton Ph.D., [1]Dr. Harry Keeling Ph.D.,

[1]Department of Computer Science, [2]Department of Education, Howard University, Washington, DC, USA

**Abstract** - *This paper proposes a novel adaptive learning system – Structured Adaptive Individualized Learning System (SAILS), which incorporates several educational theories to help engage, motivate, and cognitively stimulate students throughout an introductory computer science course. SAILS will provide remedial material as well as interventions, specifically geared towards the specific user, and periodically assess students to measure the knowledge gained. Incorporated in SAILS is a novel concept called, 'Smart Content'. Based on the student's profile and the student's proven successful grasp or lack of understanding of the topic presented, the content gets "smarter" by updating particular stored features associated with the content. The student's profile will be updated and captured as a point of reference for other similar students to determine the content features that should be updated. SAILS will provide content to similar students by analyzing the content features and past student's profile. By comparing current profiles and content features, the system would adjust the content the student receives based on its ability to detect when a student has truly mastered a specific topic or when additional practice is required in areas where the student is weak.*

**Keywords:** Adaptive Learning System, Smart Content, Preferred Learning Styles, Computer Science Education, Psychometrics, Bloom's Taxonomy, Learning Analytics

## 1. Introduction

The steady growth of technology would lead one to assume that the computer science pipeline would be overflowing with new interested recruits, and computer science programs would be graduating hundreds of thousands of students annually. The graduation rates, however, show this is not the case [1, p. 429]. One of the possible causes is that a large number of computer science (CS) students struggle and fall behind in their course work, due to a lack of prior computer science exposure and a lack of understanding of course content [2, 3]. This has led to entry, engagement, retention, and graduation issues in university computer science departments across the country [4, 5, 6, 1]. The ultimate goal of learning is for one to analyze a new situation and transfer knowledge from a past or similar situation to that new problem and develop a solution. This often does not happen in CS [2, 5]. The phenomenon of "knowledge transfer" is an essential part of the learning process. Knowledge transfer happens when a student is introduced to a new concept and the student gains a strong enough understanding of the material to be able to either apply that knowledge to a different topic or is able to apply the information to a future issue/problem. CS students are not, sufficiently grasping and understanding the computer science coursework, and adequately transferring their knowledge of foundational concepts to the application phase of learning [5]. Educators, therefore, cannot inspire and cultivate innovative minds to succeed in a competitive world [7, 8] when their students continually confused about content and lack the fundamental foundation and confidence needed to be successful computer scientists. Addressing these issues must be done in a way that makes computer science cognitively stimulating, enjoyable, more motivational and easily understandable for all interested students. Adaptive learning systems were created to be a personalized tutor for the community of learners [9, 10] and can address engagement issues. Although adaptive learning systems are currently in use in some academic settings, to completely appreciate the full potential of these systems, they need to expand on theoretical and practical concepts from the education field and accurately track the student's learning progression. One theoretical concept from education that would be useful for creating content and assessment in CS is categorization using Bloom's Taxonomy.

### 1.1 Cognitive scaffolding using Bloom's Taxonomy

Bloom's Taxonomy [11] was developed to help teachers and instructional designers guide students through lower order thinking skills up to higher order thinking skills. Activities developed using Bloom's revised taxonomy are based on five categories (remember, understand, apply, analyze, evaluate, and create), and associated objectives that students should navigate through in order to reach higher order thinking skills. In order to effectively develop those higher order thinking skills, content must be taught in a way that is meaningful to the students.

Meaningful teaching should not be haphazard. It should be associated with developing learning outcomes for lectures, activities, and tasks that are relevant to the students and their interests [12, 13]. Meaningful learning involves relating new information to existing information [14, 15, 13] and should not involve rote memorization. Meyers [16] describes three strategies for creating meaningful learning: 1) assess early, assess often – to gauge what students know; 2) let students get their feet wet; 3) welcome student input for your content and assignment. These three strategies also help to scaffold the students by gauging what they know, allowing them to actively participate in their learning by applying what they have learned, and letting them explain any unclear concepts or steps throughout the process. In turn, this will help increase self-efficacy as well as challenge the learners when they start incorporating the high order thinking skills. Utilizing Bloom's Taxonomy will also help instructors develop assessment questions for students at their particular level on the taxonomy.

### 1.2 Cognitive scaffolding in computer science

Instructors may believe that a student is incapable of learning computer science. In fact, the issue could be the student just does not have the required background information needed to grasp the concepts being taught. This could be due to a lack of exposure leading to the student misunderstanding. Thus, the accurate conclusion is the material presented did not match his cognitive scaffolding level. Johnson and Fuller [17] conducted a study of categorization of modules in computer science assessments using Bloom's Taxonomy and found there were disagreements about the level of categorization of the modules. There was difficulty because it was not clear whether the material being assessed in the study was in fact taught at the particular level on Bloom's taxonomy. The authors believe that the focus of assessment in computer science appeared to be at the application level. However, they believe the equivalent application level for computer science might be more critical than other fields that evaluate at the same level on Bloom's Taxonomy. In [18], Thompson et. al. provides a detailed analysis of Bloom's Taxonomy and its application in introductory computer science exams. They found difficulty mapping programming questions to the description of the original taxonomy. At the end of the study the authors found that it was beneficial to have the instructors who taught the course involved in the categorization of exam questions. Much work still needs to be done in the area of placing content and assessments in correct categories that are on students' cognitive levels and associate levels of difficulty for the categories. The focus should be to scaffold and build the student throughout the course in the areas that will be assessed. It should also be clear what category each of the assessment questions belongs, as well as which category the remediation content being covered belongs on Bloom's Taxonomy. The categorization of content and assessments will help in developing an learning situation that is manageable and meaningful to the student. It is important that the process of acquiring knowledge and evaluating true learning be meaningful and properly guided to ensure that the knowledge is stored in the student's long term memory [19, 20].

### 1.3 Psychometrics

Psychometric evaluation is used as a way to assess questions to ensure the scores created from using the assessment are valid and reliable. Questions that have been psychometrically evaluated also ensures that the questions are fair to the students taking the assessment and not biased. The idea behind knowledge acquisition in computer science is to ultimately be able to think critically and solve problems. Critical thinking should incorporate transfer of knowledge [15, 21] from one computer science related situation to another in order to solve the problem [22]. This involves finding the link between past situations and the possible application it has to the current situation. However, because knowledge is a trait that cannot be fully observed, it is difficult to identify if a student has acquired the knowledge when content has been taught. Employing psychometric assessments in an adaptive system is a way in which a professor can better measure knowledge acquisition. Psychometrically-sound assessments will help in determining knowledge acquisition, because the assessments have questions that reliably and validly measure specific knowledge [23]. It is important to assess students in this way to help determine where they are and what should be done to get them where they should be at certain check points in a course. Many times concepts in computer science are taught, but it is unclear whether knowledge has truly been attained nor whether the assessment is a true gauge the acquired knowledge. Knowledge acquisition and valid and reliable assessments are vital to helping students with knowledge transfer and usher them through the computer science pipeline and into successful career paths. This paper discusses some theories that would be beneficial to use in computer science classes and presents an adaptive learning system that will incorporate them. Section two of this paper gives a literature review on existing adaptive learning systems with preferred learning styles and the challenges that currently exist. Section three will discuss Cognitive Scaffolding using Bloom's Taxonomy and psychometric assessment (using valid and reliable assessments to measure knowledge). Section four will cover SAILS.

## 2. Adaptive learning

Adaptive learning systems are software systems that present different content to learners in response to the student's performance. Being effective and efficient computer science faculty involves having the ability to assess and manage each students' cognitive development. Faculty members are expected to take time to make the entire learning experience meaningful for the student, answer student questions, and address concerns in real time. If these expectations are not met the effectiveness of learning diminishes [24]. Properly identifying each student's varying needs can be overwhelming to keep track of. The reality is that most courses reduce to a “one-size fits all” approach. Adaptive learning systems can reduce the work load of faculty while increasing the individualization of students' learning [25] by changing content and delivery for a user based on their learning profile [7]. By using an adaptive learning system the teacher would not have the arduous task of, knowing each student's area of struggle, their current knowledge level on various content, and being cognizant of students' varying approaches to learning.

Adaptive learning systems have gone by several names (i.e. Adaptive Hypermedia System (AHS), Adaptive Educational Hypermedia Systems (AEHS), and Technology Enhanced Learning Environment (TELE)) [26, 27, 28, 9] and have been partially driven by a realization that tailored learning cannot be achieved on a large-scale using traditional, non-adaptive approaches [29]. One approach to adapting learning to a user is based on their learning styles.

## 2.1 Preferred learning styles and their use in adaptive systems

Adaptive systems that use learning styles also employ various techniques to adapt to a student model [30, 10], such as, fragment sorting [31], customized system interface, and adaptive selection of learning objectives [32]. Kolb [33] believed that learning should be looked at in terms of the process involved instead of the behavioral outcome expected. For example, ideas from the experiential learner, are formed and reformed as opposed to fixed and immutable elements of thought. With the apparent differences in the way students learn, using an adaptive system that provides content based on preferred learning styles both personalizes and enhances the learning experience for the student by presenting material in a manner that is more appealing and effective [10]. It gives the students the opportunity to participate in their learning which in turn will reinforce the knowledge they have acquired. Preferred learning styles are the ways in which a learner prefers to receive and grasp content best. There are over 70 learning style assessments developed [26]. Although adaptive systems using preferred learning styles is not a new concept [28, 9], it still remains widely underutilized. However, not everyone agrees with the concept of preferred learning styles [34]. Rohrer and Pashler [35] state that there is no sufficient evidence (validity) to prove the learning styles truly exist. Contrary to Pashler et. al.'s claims [34, 35], Felder and Spurlin [36] describe statistical methods used to analyze and validate the Felder-Silverman model. In addition, the authors identify what the Felder-Silverman Index of Learning Styles is, as well as, what are the uses and misuses of the model. Felder and Spurlin also point out that the learning styles are not reliable in terms of indicating strengths and weaknesses. There are four dimensions of the Felder-Silverman Learning Style Index, and of the four dimensions, Hwang et. al [37] used the Sequential-Global dimension in an educational computer game. They had two versions of the game and provided questions to students in the study. A sequential learner is one that prefers information presented in linear and orderly steps; they prefer step-by-step in order to see the big picture. Global learners prefer information present as a whole; they want to have the big picture in order to solve the problem. The questions presented in Hwang et. al.'s games were related to either the sequential or the global styles to ascertain the student's learning style. Based on the results of the questions they presented either the sequential or global version of the game to the student. They found that students learned best when presented with the version of the game that was based on their learning style. The concept of developing a system that can adapt to learning styles serves as a great tool to handle student differences, however, the current systems still fall short as they are not able to determine if knowledge has been acquired nor if knowledge transfer has occurred.

## 2.2 Remediation and adaptive learning systems

When students in computer science get tutored, a human tutor applies different approaches depending on the student's focus area to ensure that the student understands the concepts being covered. However, in a classroom setting the remediation approaches for all students who find themselves falling behind are generally the same, which may result in loss of interest from some students. Outside the classroom, the same reading materials are recommended, and the same assessments are given to all students no matter what foundational information is needed by the individual student. Therefore, all concepts are covered in the same way to every student. This "one size fits all" remediation approach has not been proven to be effective. While some students catch up with the rest of the class, many others remain unengaged. Those who remain unengaged by this remediation style continue to fall further behind [10]. Remediation should be tailored and administered with consideration of the students' differences.

## 2.3 Challenges and possible solutions

Various differences in individuals require diverse methods of acquiring knowledge and dissemination of the knowledge [38, 39, 40]. After reviewing the literature we noticed that some existing gaps must be addressed. To mitigate the gaps, an effective adaptive learning systems would immensely benefit the student if: 1) the content is presented to the student based on their learning style; 2) the content presented is evaluated to ensure that true transfer of knowledge is taking place and continues to take place throughout the student's interaction with the system; and 3) the remediation and intervention content has been correctly categorized and tailored the student. This will help students as well as instructors tailor remediation efforts for each student. To mimic the remediation efforts of human tutors, adaptive systems should be tailored to the specific skill level of the individual student. This means covering applicable content that is needed for that specific student, and presenting it in the student's particular learning style. This form of remediation should only be applied to other students when there are enough similarities in the student profiles. To ensure successful interaction and effective learning, sufficient metadata needs to be collected and assigned to the content and the student's profile. In the following sections, we give a review of educational components that are important to address the gaps and how those components are currently being implement in computer science.

# 3. SAILS – Structured Adaptive Individualized Learning System

A Structured Adaptive Individualized Learning System (SAILS) is a proposed supplemental system to classroom instruction. SAILS will serve as a means to adapt to students' current learning level and ability and guide each student through high order thinking in an introductory computer science course. The ultimate goal of SAILS is to shift some of the workload off the teacher, and focus on ensuring that proper scaffolding of computer science concepts is taking place for each individual student. At the commencement of the course, each student will take the Felder-Silverman Learning Style Index to determine his/her preferred learning style. Throughout the course, material (remedial, intervention, and general conceptual information) will be presented to the student primarily based on his/her assessed learning style and additional supplemental material will be presented using least preferred learning styles.

## 3.1 Preferred learning style in SAILS

Felder believes in order to build mental dexterity it is vital that students learn based on their preferred learning style coupled with a least preferred learning style in order to become well rounded learners [41]. This is called, "teaching around the cycle". When presenting remediation/intervention material, SAILS will present the content using the student's preferred learning style coupled with at least one least preferred style to keep with the concept of "teaching around the cycle". When a student is aware of his/her learning style they are more confident about how to best tackle a new or challenging topic [42]; therefore SAILS will provide students with the results of his/her assessed Felder-Silverman Learning Style Index. According to Felder the more aware a teacher is of the learning style preferences of students, the better equipped (s)he will be to effectively convey the material. Learning how to benefit from the differences in their own learning style is important to the student's academic success.

## 3.2 Remediation vs intervention

The goal of SAILS is to provide dynamic individualized paths to success by presenting personalized interventions. The distinction between remediation and intervention is the point at which the material is presented. Remediation material is provided to students who are behind or failing. Intervention material is provided as a way to intervene before the point of failure. Providing interventions early enough will help build students' self-efficacy, therefore it is important to collect sufficient data to determine when intervention is needed.

## 3.3 Valid assessments

SAILS will incorporate psychometrics by developing various psychometrically-sound assessment questions associated with each cognitive scaffolding category. Implementing an adaptive learning system that incorporates psychometrically evaluated questions will serve as a way to measure whether the computer science concepts are truly being transferred in the way that is intended. This will help easily determine the student's skill level and the knowledge that has been acquired. Each assessment item will have an associated level of difficulty as well as a designated category on Bloom's Taxonomy. This will help steer students through successful intervention paths and the remediation of each student.

### *3.3.1 Cognitive scaffolding*

In an effort to properly scaffold the learner, all materials in SAILS (assessment questions and remedial/intervention) will be evaluated and placed in a specific cognitive category on Bloom's Taxonomy. The system will evaluate the learner's profile and present the appropriate material to the learner. This will ensure that the students are learning and being assessed based on their own cognitive level and will be able to end the semester on the expected level based on the learning outcomes of the course.

# 4. Conclusion and future work

The goal of computer science is to get students to eventually create and contribute to the field. SAILS aims to build/scaffold the student so that by the end of a course they will have the ability, confidence, and willingness to create and contribute to the field of computer science. Content in SAILS will have metadata representing features that will be captured and updated frequently and used to scaffold similar students.

## 4.1 Adapting via Smart Content

SAILS will be adapting and presenting material to students using 'Smart Content'. All content in SAILS will have metadata associated with it and a score that will help make the content smarter. Smart Content will aid in the analytics and proper distribution of content to appropriate student profiles. During the commencement of the data collection process, SAILS will focus on distributing remediation materials to the students who are failing. The student will then be assessed after they have reviewed the remediation materials presented to them. The metadata associated with the remediation material will be analyzed. The outcome of the analysis will determine whether the material was effective or not for that student. The metadata will then be updated and depending on the score, will be an indication of whether or not the remedial material will be distributed to a student with a similar profile. Once the system has been trained and has enough analytics to support distribution of material to students given a particular profile, interventions will decrease the need for remediation. SAILS will make use of the collected analytics and flag students who are in a set danger zone. To help build student self-efficacy in the course, the system-designated interventions will be provided to the student before he/she reaches the point of failure. These interventions will be supplemental multimodal material presented in the student's preferred learning style, their current category on Bloom's Taxonomy, and periodic validated assessments that will help determine knowledge of the topic. These interventions will be presented to the student based on the intervention's proven ability to improve other similar students' success paths. The interventions provided to the student will build/scaffold the student where they are on Bloom's Taxonomy and take them through to the higher order thinking skills. The analytical data will help steer future students through successful intervention paths, and reduce possible need for remediation. The result of SAILS will help the professor get a clearer view of individual and overall students' weak topic areas. The abstract nature of computer science will make this challenging, but not impossible. SAILS' use of Smart Content, psychometrics and Bloom's Taxonomy would benefit both the teacher, by helping the teacher set questions that measure the knowledge acquired at the students' current level, and the student by helping to truly guide their learning. This will help to ensure that the assessment questions in each category are reliable, fair, and valid and result in the student's success in the course.

## 5. REFERENCES

[1] National Science Board, "Science and Engineering Indicators 2014," National Science Foundation (NSB 14-01), Arlington, VA, 2014.

[2] M. Haungs, C. Clark, J. Clements and D. Janzen, "Improving First-Year Success and Retention Through Interest-based CS0 Courses," in *SIGCSE'12*, New York, NY, 2012.

[3] W. Aspray and A. Bernat, "Recruitment and Retention of Underrepresented Minority Graduate Students in Computer Science," Computing Research Association, Washington, DC, 2000.

[4] P. Kinnunen and L. Malmi, "Why Students Drop Out CS1 Course?," *ICER '06,* vol. 5, no. 3, pp. 97-108, 2006.

[5] R. A. Dixon and R. A. Brown, "Transfer of Learning: Connecting Concepts During Problem Solving," *Journal of Technology Education,* vol. 24, no. 1, pp. 2-17, 2012.

[6] T. L. Crenshaw, H. Metcalf, E. W. Chambers and U. Thakkar, "A Case Study of Retention Practices at the University of Illinois at Urbanna-Champaign," in *SIGCSE*, Oregon, 2008.

[7] J. Froyd and N. Simpson, "Student-Centered Learning Addressing Faculty Questions about Student-Centered Learning," in *Course, Curriculum, Labor, and Improvement Conference*, Washington, DC, 2008.

[8] G. A. Gabriele, "Advancing Engineering Education in a Flattened World," *Journal of Engineering Education,* pp. 285-286, 2005.

[9] P. Brusilovsky and C. Peylo, "Adaptive and Intelligent Web-based Educational Systems," *International Journal of Artificial Intelligence in Education,* vol. 13, pp. 156-169, 2003.

[10] E. Popescu, C. Badica and L. Moraret, "Accommodating Learning Styles in an Adaptive Educational System," *Informatica,* vol. 34, pp. 451-462, 2010.

[11] D. R. Krathwohl, "A Revision of Bloom's Taxonomy: An Overview," *Theory Into Practice,* vol. 41, no. 4, pp. 212-265, 2002.

[12] D. G. Kipnis and G. M. Childs, "Educating Generation X and Generation Y: Teaching Tips for Librarians," *Medical References Services Quarterly,* vol. 23, no. 4, pp. 25-33, 2004.

[13] D. B. Barron and D. L. Darling-Hammond, "Teaching for meaningful Learning: A Review of Research on Inquiry-Based and Cooperative Learning," The George Lucas Educational Foundation, 2008.

[14] J. Ormrod, "Learning, Cognition, and Memory," in *Essentials of Educational Psychology*, Pearson Prentice Hall, 2006, pp. 16-53.

[15] J. E. Ormrod, "Human Learning," in *Human Learning*, Boston, Pearson, 2012.

[16] S. A. Meyers, "Three Strategies for Creating Meaningful Learning Experiences," Faculty Focus, 24 02 2014. [Online]. Available: http://www.facultyfocus.com/articles/effective-teaching-strategies/three-strategies-creating-meaningful-learning-experiences/.

[17] C. G. Johnson and U. Fuller, "Is Bloom's Taxonomy Appropriate for Computer Science?," in *Proceedings of the 6th Baltic Sea conference on Computing education research: Koli Calling 2006*, 2006.

[18] E. Thompson, A. Luxton-Reilly, J. L. Whalley, M. Hu and P. Robbins, "Bloom's Taxonomy for CS Assessment," in *Thenth Australasian Computing Education*, Wollongong, Australia, 2008.

[19] D. Shaffer, W. Doube and J. Tuovinen, "Appying Cognitive Load Theory to Computer Science Education," in *Workshop of the Psychology of Programming Interest Group*, 2003.

[20] M. Taylor, "Teaching Generation NeXt: A Pedagogy for Today's Learners," The Higher Learning Commission, 2010.

[21] J. E. Ormrod, Educational Psychology: Developing Learners, Seventh ed., Boston: Pearson, 2011.

[22] D. F. Halpern, "Teaching Critical Thinking for Transfer Across Domains," *American Psychologist,* vol. 33, no. 4, pp. 449-455, 1998.

[23] F. B. Baker, The Basics of Item Response Theory, ERIC, 2001.

[24] R. A. Berk, "Teaching Strategies for the Net Generation," *Transformative Dialogues: Teaching and Learning Journal,* vol. 3, no. 2, pp. 1 - 24, November 2009.

[25] Education Growth Advisors, "Learning to Adapt: A Case For Accelerating Adaptive Learning in Higher Education," 03 2013. [Online]. Available: http://edgrowthadvisors.com/wp-content/uploads/2013/03/Learning-to-Adapt-White-Paper_Education-Growth-Advisors_March-2013.pdf.

[26] C. Mulwa, S. Lawless, M. Sharp, I. Arnedillo-Sanchez and V. Wade, "Adaptive Educational Hypermedia Systems in Technology Enhanced Learning: A Literature Review," in *SIGITE'10*, Midland, Michigan, 2010.

[27] E. Verdu, L. M. Regueras, M. J. Verdu, J. Pablo De Castro and M. A. Perez, "Is Adaptive Learning Effective? A Review of the Research," in *Applied Computer & Applied Computational Science*, Hangzhou, China, 2008.

[28] P. Brusilovsky and E. Millan, "User Models for Adaptive Hypermedia and Adaptive Educational Systems," in *The Adaptive Web*, Berlin, Springer-Verlag, 2007, pp. 3-53.

[29] L. Nguyen and P. Do, "Learner Model in Adaptive Learning," *World Academy of Science, Engineering and Technology,* no. 21, pp. 395-400, 2008.

[30] P. Brusilovsky, "Methods and techniques of adaptive hypermedia," *User Modeling and User Adapted Interaction,* vol. 6, no. 2, pp. 87-129, 1996.

[31] P. Brusilovsky, "Adaptive Hypermedia," *User Modeling and User-Adapted Interaction,* vol. 11, pp. 87-110, 2001.

[32] E. Triantafillou, A. Pomportsis and S. Demetriadis, "The Design and the Formative Evaluation of an Adaptive Educational System Based on Cognitive Styles," *Computers & Education,* vol. 41, pp. 87-103, 2003.

[33] D. A. Kolb, Experiential Learning: Experience As The Source Of Learning And Development, Englewood Cliffs, New Jersey: Prentice Hall, 1984, pp. 20-38.

[34] H. Pashler, M. McDaniel, D. Rohrer and R. Bjork, "Learning Styles: Concept and Evidence," *A Journal Of The Association For Psychological Science,* vol. 9, no. 3, pp. 105-119, 2009.

[35] D. Rohrer and H. Pashler, "Learning Styles: Where's The Evidence?," *Medical Education,* pp. 634-635, 2012.

[36] R. M. Felder and J. Spurlin, "Applications, Reliability and Validity of the Index of Learning Styles," *International Journal of Engineering Education,* vol. 21, no. 1, pp. 103-112, 2005.

[37] G.-J. Hwang, H.-Y. Sung, C.-M. Hung and I. Huang, "A Learning Style Perspective to Investigate the Necessity of Developing Adaptive Learning

Systems," *Educational Technology & Society,* vol. 16, no. 2, pp. 188-197, 2013.

[38] A. Alharbi, D. Paul, F. Henskens and M. Hannaford, "An Investigation Into The Learning Styles And Self-Regulated Learning Strategies For Computer Science Students," in *ASCILITE*, Australia, 2011.

[39] C. Zander, L. Thomas, B. Simon, L. Murphy, R. McCauley, B. Hanks and S. Fitzgerald, "Learning Styles: Novices Decide," in *ITiCE'09*, Paris, France, 2009.

[40] M. Keeley, "The Basics of Effective Learning: Knowing Yourself as a Learner," Bucks County Community College, 10 5 2011. [Online]. Available: http://faculty.bucks.edu/specpop/knowing.htm.

[41] R. Felder, "Matters of Style," *ASEE Prism,* vol. 6, no. 4, pp. 18-23, 1996.

[42] R. M. Felder and L. Silverman, "Learning and Teaching Styles in Engineering Education," *Engineering Education,* vol. 78, no. 7, pp. 674-681, 1988.

# Using Problem-Based Learning in a CS1 Course
## -Tales from the Trenches-

**R. M. Kaplan**[1]
[1]Department of Computer Science and Information Technology, Kutztown University of Pennsylvania, Kutztown, PA, United States

**Abstract** - *Putting learning into the hands of learner, making learners responsible for there own learning, and flipping the classroom are all terms you hear today to describe different approaches to teaching various subject matter to students. The idea that students will embrace the idea of responsibility for their own learning presumes that (a) students are ready to do so and (b) students have a desire to learn. In the Fall of 2014, a unique opportunity presented itself in the form of being able to teach two sections of the first core course, CS1, in the computer science curriculum. This unique opportunity would provide a basis for comparing the two standard teaching methods for teaching programming to novices. The traditional lecture-based approach and problem-based learning (PBL) could be compared. The creation of problems or cases is fundamental to the PBL approach to teaching. Finding cases that can be used or adopted into use is not possible because of the complete lack of exemplars. The purpose of this paper is to describe three of the problems/cases used in the PBL version of the CS1 course and to also make some preliminary observations about using PBL in the classroom.*

**Keywords:** Computing Education, Computing Education Program, CS1

## 1 Introduction

It has long been known that one of the most difficult classes a student will take when majoring in computer science is their first programming course[4]. Programming is a difficult skill to learn because it entails learning several different kinds of knowledge and being able to employ them at the same time[1]. When one learns to program, one must be able to solve problems. If an individual does not have a well-honed ability to solve problems, chances are that this individual will not be a good candidate for programming. Above all else, problem solving is key to be successful in programming. Years ago, Edsgar Dijkstra wrote a lecture entitled, “On the cruelty of really teaching computer science[3].” In this lecture he makes the following statement.

“… my contention that automatic computers represent a radical novelty and that only by identifying them as such can we identify all the nonsense, the misconceptions and the mythology that surround them. Closer inspection will reveal that it is even worse, viz. that automatic computers embody not only one radical novelty but two of them. “

Dijkstra’s observation was stating that regardless of the fictions that were told about computers and using computers, learning how to carry out tasks using computers was extremely difficult.

Language skills are another aspect of learning to program computers that add to the complexity of the task. We still communicate with computers in extremely primitive languages (as compared to English). This downshifting is problematic because human beings, once instructed in the marvelous expressiveness of English find it very difficult (at least initially) to express their thoughts in “computer-ease.” Taken together a new programmer must learn to solve problems, write the solution in a special form called an algorithm (still in English), rewrite the algorithm into computer language, and then attempt to correct problems in what they have written to actually make the computer carry out the instructions in the program they have written – specified in a language that is largely unknown to the student. That would be like asking a young person learning English to grammatically correct a college-level paper. They would not have the vocabulary nor would they have the grammar rules to do so.

## 2 Background

This study was carried out at Kutztown University, one of 14 state schools in Pennsylvania. The university is a rural university as the community it resides in is much more rural then urban. As such many of the students that attend the university may be first generation attendees meaning the first person in their immediate family to attend an institution of post-secondary education. The computer science department at the university is a small one consisting of 11 full time faculty. The department does not use adjuncts for courses. Full time faculty teaches all courses. Many of the faculty have significant non-academic experience including industry, experience. Some of the faculty has extensive experience in designing and implementing systems. The department has approximately 200 full time students. This number has stayed fairly constant over the past 5 years.

A student in the department can earn a degree in computer science. They can specialize in Information Technology or Computer Science. For the first two years the programs are similar. In the third and fourth year the programs diverge and students can specialize in their specific programs (computer science or information technology). All students must take the basic programming courses consisting of what is known in computer science programs Computer Science I and Computer Science II (CS1, CS2). In order for a student to continue on to other courses in both programs, CS1 and CS2 must be passed (C or better). Specifically it is necessary for a student to complete CS2 which in turn requires CS1.

In CS1 students learn the fundamental concepts that are required to write rudimentary programs. The students are required to write six programs that are specifically defined for them. The student must write an algorithm to solve the problem defined in specification for the program, write the code for that algorithm, correct any syntax errors, and run the program and correct any run time errors. The students are taught how to use command line tools in UNIX and are required to write all of their programs with these tools. EMACS is used as the editor and the GNU C++ compiler is also used. CS1 does not cover object-oriented concepts.

The concepts that are presented in the course are as follows:

Basic editing of a source file using Emacs
Compiling a program (and what that means)
Running a program (and what that means)
The basic structure of a C++ program
Constants, variables, simple console output, arithmetic, simple console input, conditional expressions and conditional statements (including the switch statement), iteration, file input/output, and arrays.

# 3 Common Characteristics of Both Versions of CS1

The CS1 course is taught over a period of 15 weeks. Each week the students can expect to meet three times a week for 1 hour or two times a week for 90 MINUTES. Approximately every 2-3 weeks the student will get a programming assignment that will be due in 1 week (if easy), and 2 weeks (if moderate). The class ends with a common examination that is given to all sections of the course that are going on during the term (in some terms there may be 4 sections of the CS1 class).

## 3.1 Characteristics of the Traditional CS1 Class

In addition to the common characteristics of both classes, the traditional CS1 class involves traditional lectures, in-class worksheets and exercises, and programming assignments. Some interaction between students and instructor may occur in a question/answer setting. Each of the projects is more complex than the last and makes use of newly taught concepts and constructs. By the sixth program students should be able to use arrays in the programs they write.

## 3.2 Characteristics of the Problem-Based Learning (PBL) CS1 Class

There are striking and marked differences between the traditional CS1 class and the PBL class from the very beginning of the class. The PBL methodology as defined by Nuutila, Torma, Malmi[3] consists of the seven steps listed in table 1.

| |
|---|
| Opening session – half an hour, in the group |
| Step 1: Examination of the case. The group gets familiar with the case material.<br>Step 2: Identification of the problem. An initial title for the case is specified.<br>Step 3: Brainstorming. The students present their associations and ideas about the problem to find out what is already known and how does the case relate to the previous knowledge. The ideas are said aloud and written on self-stick notes, which are organized on a white board.<br>Step 4: Sketching of an explanatory model. An initial version of the explanation for the problem is constructed and most important concepts and their relations are identified.<br>Step 5: Establishing the learning goals. Those parts of the explanatory model that are mysterious, fuzzy, or simply unknown are identified and the central ones are chosen as learning goals for the group. |
| Study period -- one week, each student working independently |
| Step 6: Independent studying. Each student independently studies to accomplish *all* learning goals. This phase includes information gathering and usually a substantial amount of reading (e.g., 50--150 pages). |
| Closing session -- one to two hours, in the group |
| Step 7: Discussion about learned material. Equipped with the newly acquired knowledge, the group reconvenes to discuss the case. The discussion includes *explanation* of central concepts and mechanisms, *analysis* of the material, and *evaluation* of its validity and importance. |

Table 1 – Seven Steps of PBL[3]

Table 1 is shown here because it is a concise specification of the PBL-process. Notice in Step 2 that students identify the problem. PBL is based on the idea that students are given an ill-formed problem to solve. The problem may be missing information, specify the wrong data, or state incorrect assumptions. Students must work through the problem in order to formulate a solution. Students work in teams. Each team is responsible to come up with a problem as a team and then each individual is required to submit own their problem solution. One of the first problems that the students must solve is to define the criteria to pick the individuals that will comprise their group. They must establish criteria for the leader and then use these criteria to select the each person in their group. The problem statement is shown in Figure 1.

**PROBLEM 0 Problem-Statement**

The first problem for you to solve is the problem of forming groups. In a typical setting, your instructor would assign everyone to a group or the instructor would ask that you assign yourselves to a group. There are many problems with these approaches. Although these approaches are efficient they do not typically result in the best groups and most effective group organization.

Problem 0 asks you to come up with a process to select members of groups. A group consists of 5 to 8 individuals. The goal for your process is to assemble the best group of individuals possible – i.e., groups that will be more effective that those that might be arrived at by the approaches above. In order to come up with such a process it is necessary to understand how you might assess the membership of a group to determine if the group formed by the process formulated as the solution to problem 0 is better than groups that would be formed otherwise. It is not only a selection process but also an assessment process to determine how much better assembled group performance will be.

Problem Statement: How can you determine group population from a diverse group of individuals and insure that the performance of the selected group will be better than groups formulated with traditional means.

Figure 1 – First PBL Problem (Introduction to PBL)

The purpose of this problem was to get the student's "feet wet" in this mode of learning – to give students experience as it were. Each problem was formulated with a description and also a problem statement defining the problem that has to be solved.

One of the first things that you learn when using the PBL method is that you must schedule significantly more time for the students to work on problems. The problems are formulated to provide a basis for concepts that are necessary for the novice programmer to be successful.

Ill-formed problems are important because they require students to determine what is "wrong" with the specification of the problem. For example, in Problem 0 there is significant information missing about how to form groups from a larger group. No process is given and the students have to formulate their own process. Two examples of student processes that solve this problem are shown in figures 4 and 5.

**Counting Out Change**

Just in case you need a vocation to fall back upon, I am going to describe how you count out change. If you ever need to get a job as a cashier with a backup plan you will be able to do so without any problem if you can master the following.

Supposing that I have purchased $5.36 worth of items in a convenience store. I give the store clerk a $10 bill. The store clerk rings up the sale, the cash drawer opens and the cashier begins counting out change.

First come the pennies – $5.37, $5.38, $5.39, $5.40. (4 pennies)
Next come the dimes or the nickels – If nickels $5.45, $5.50. (2 nickels)
Third comes the quarters – $5.75, $6.00 (2 quarters)
And lastly comes the dollars - $7.00, $8.00, $9.00, $10.00 (4 dollars)

So the cashier gives to me $4.64 comprise as:

4 dollars 2 quarters 2 nickels 4 pennies

Here is Problem 2.

The people that you train to be cashiers have to know how to count change. And you don't care one way or another if the cash register can actually tell the cashier how much change someone is supposed to get back.

How would you describe a process to count change back to a customer when the purchase is an arbitrary amount, the amount tendered is arbitrary and of course the change back is dependent on the amount spent? Basically you are to describe a process by which a cashier can determine how many $10 bills, $5 bills, $1 bills, Quarters, Dime, Nickels, and Pennies should be given back to the customer as change.

Remember, what ever process you describe must be of sufficient detail to be doable by an untrained (as of now) cashier.

Figure 2 – Second PBL Problem (Algorithm Writing)

**Ardmore and Narberth**

Narberth and Ardmore, two brothers, decided to go into business together. They thought they would be able to create an excellent and successful restaurant. They were very excited. They remodeled their restaurant space, put in all of the necessary cooking tools, prepared a menu, and got ready to open. On one sunny Saturday they opened for lunch. And then some unexpected things happened.

Customers came into their restaurant out of curiosity because they know it was a new restaurant. In that way, business was good. Customers would be seated, and handed a menu, which looked pretty typical of restaurant menus.

For lunch the restaurant served a good assortment of sandwiches and soups. In addition for lunch, there were a couple of warm entrees and also some desserts. Drinks were also on the menu.

So far so good. Everything seemed to be normal, and the customers were looking forward to their lunch.

Much to the patron's surprise, service was very quick. Trays came out of the kitchen piled high with lunch, or so they thought.

One of the patrons ordered a swiss mushroom hamburger. On the hamburger he wanted lettuce, tomato, and fried onion. He ordered a side of fries with the hamburger and a Coca-Cola to drink. What he got was not quite what he expected.
Delivered to his table were the following items.

1 uncooked hamburger patty
1 unsliced hamburger roll
1 onion
1 head of lettuce
1 tomato
3 potatoes
1 cup of coke syrup
1 bottle of seltzer

And this was delivered as the patron's lunch. The patron was, to say the least, very surprised. He inquired of the waiter, "What is this?" To which the waiter answered, "That is the lunch you ordered." The patron said, no, this is not the lunch I ordered, it is only the ingredients for the lunch I ordered. At that point the patron got up and left the restaurant. Ardmore and Narberth looked at one another in disbelief. It became worse because all of the patrons were leaving the restaurant. Within about 5 minutes the successful opening turned into a disaster and all that Ardmore and Narberth were left with was tables full of ingredients. They had no idea what went wrong. After all they were serving what people ordered.

You have been hired as a consultant by Ardmore and Narberth to help them figure out what went wrong. You have been in the restaurant business a long time and have helped many startup restaurants. You didn't think this would be a problem. You told Ardmore and Narberth to do what they did on opening day.
At the end of the ordering/delivery of the meal you knew exactly what is wrong. The problem is that the brothers didn't get it (a little thick if you wish). Both brothers are very process oriented. You knew if you came up with a process for them, that you may be able to correct the problem.

**Problem Statement:**
Determine the best way to explain to Narberth and Ardmore the process by which they should operate thereby correcting the problem in their restaurant. You need to introduce to them a concept that will make clear how they can be successful.

Figure 3 – Third PBL Problem (Functions)

To make sure that each group has the best possible chance at being successful I would choose a group based on these steps

1) Split the potential group members into sides with those who have had prior programming experience on one side, and those that have not on the other.
2) Assign (if Possible) one person who has had prior experience with someone who does not.
3) Break these teams into groups of 5-8 people that way no one is left out.

I believe that this has the best possible chance to succeed because this way of assigning group members gives each group (Probably) an even number of experienced programmers, and those who haven't done any programming before, which will allow those who have learned a bit of programming to help those who don't know that you have to put a semicolon at the end of a line while going through the Projects.

Figure 4 – Sample Solution A – Group Formation Problem

To pick the best group the entire class should write down their age, computer programming experience, age and other life experience. Life experience could be working in a corporate setting with a familiarity of result driven collaborating . Why is this the best way to pick a group? Placing together a group of expert programmers would not necessarily mean they would be the best at the task at hand. Using the strategy outlined above, a group would be able to form with a *diverse* skill set, and not just a group with the most experienced programmers.

1. Every student will write their age.
2. Every student will write what their familiarity is with programming.
3. Every student will write down other skills they have.
4. The class will have this information and pick a group accordingly.

Figure 5 – Sample Solution B – Group Formation Problem

Eighteen group selection strategies were submitted. To come up with a single strategy the students multi-voted for strategies they felt were the three best strategies. Upon choosing the best three strategies, the students then discussed each of the strategies as a potential strategy for their group selection process and decided upon the strategy they felt was best to use. They then used this strategy to choose the members of their groups. Three groups were formed with approximately six students per group. These groups were maintained for the remainder of the course.

After the group selection process was executed, the students were then asked to answer some reflective questions about the process and meta-process they followed.

Question 1: What did you learn from this process? Identify everything you learned – new concepts, new knowledge, old knowledge that was reinforced. See if you can create an inventory of your learning from this problem process.

Question 2: What did you think about the solution to the problem? Are you in agreement with the approach or are you in disagreement with the approach?

Question 3: What suggestions could you make to improve the process? What aspects would you change? What aspects would you leave as they are?

Question 4: Was there anything that you had trouble understanding? Did you bring this up in your group? Did the group try to assist you in understanding?

Student responses to these questions were then collected. To some extent, the messages represent a measure of engagement in the process. Here is one example of a student response to these questions.

**Question 1:** From the group selection process I learned things such as skill level can help guide the picking of group members. After the picking of the group members I could see how this was a helpful step in selecting members.

**Question 2:** I was not for the solution we went for at the beginning due to the fact that there was no mention of scheduling when picking group members which was a main point for me. However later when we started picking group members I quickly found myself liking the outcome and the range of people with different knowledge levels. I also liked how there was a way to assess knowledge without bringing years of practice into the mix. This was good for me because I personally don't have many years of experience with programing.

**Question 3:** The only thing that I would add to this process is some sort of assessment so that schedules and free time could at least be a part of the process. Other than that I liked this process in all aspects.

**Question 4:** There was really nothing I had an issue understanding.

# 4 Observations and Conclusions

1. PBL is a significantly more complex approach to teaching material.

PBL is not at all like the traditional approach to teaching programming using lectures, homework, tests, and projects to teach and reinforce programming knowledge. In PBL students must gain this knowledge through carefully conceived problem scenarios that lead them to learning the correct knowledge so that they may solve computer-programming problems. An instructor undertaking this approach must construct problems that meet the objectives of the course. Such constructions require a significant amount of time both to conceive and to execute.

2. PBL takes a longer time to teach the same material as does the traditional approach to teaching the same material.

In order to teach the skills necessary for programming in PBL one must devise analogical problems that will give students insights into the skills they will need to write computer programming. For example, in the counting out change problem students need to devise a way to explain how to how to count change to a person who can only understand extremely well specified steps. This problem gives students insights and knowledge about algorithm writing. Students are required to write algorithms for each programming project they are given. In terms of the seven step approach elaborated by Schmidt[5] and implemented in a programming course by Nuutila, Torma, and Malmi [3] it is clear that this process would require more time than the traditional approach to teaching CS1.

3.Students may not "buy into" the idea of teaching themselves.

The current "crop" of students (at least in the United States) seem to be less interested in being responsible for their knowledge acquisition then having someone else be responsible for providing all of the information to them. The result of this is that students do not necessarily agree with this approach to teaching.

4. There are very few documented scenarios of applying PBL to CS1, i.e., there is not a large body of problems to draw upon.

Other than Nuutila, Torma, and Malmi[3] there are few examples in the literature that show PBL-type problems they have used in a PBL-based approach to teaching CS1. Although PBL is not a new approach to teaching, it still seems that such an approach may yield beneficial results as compared to traditional approaches to teaching CS1. More examples are needed of sample curriculums that take this approach.

# 5 References

[1] Bennedsen, J. and M. E. Caspersen (2005). Revealing the programming process. Proceedings of the 36th SIGCSE technical symposium on Computer science education. St. Louis, Missouri, USA, ACM**:** 186-190.

[2] Dijkstra, E. W. (1989). "On the cruelty of really teaching computing science." Communications of the ACM **32**: 1398-1404.

[3] Nuutila, E., et al. (2005). "PBL and computer programming—the seven steps method with adaptations." Computer Science Education **15**(2): 123-142.

[4] Sarpong, K. A.-m., et al. (2013). "Causes of Failure of Students in Computer Programming Courses: The Teacher Learner Perspective." International Journal of Computer Applications **77**(12): 27-32.

[5] Schmidt, H. G. (1983). "Problem-based learning: rationale and description." Medical Education **17**: 11-16.

# How to Develop Competencies and Abilities Professional in Software Engineering in Undergraduate Students?

**POSITION PAPER**

Carlos S. Portela, Alexandre M. L. Vasconcelos
Informatics Center (CIn)
Federal University of Pernambuco (UFPE)
Recife, Brazil
{csp3, amlv}@cin.ufpe.br

Sandro R. B. Oliveira
Postgraduate Program (PPGCC)
Federal University of Pará (UFPA)
Belém, Brazil
srbo@ufpa.br

*Abstract*—**How many Software Engineering (SE) topics are seen during undergraduate courses? How many of these are relevant for the professionals in the area? We are analyzing the amount and relevance of Software Engineering topics recommended by the curriculum guidelines of ACM/IEEE and of the Brazilian Society of Computer (SBC). In addition, we intend to analyze the effectiveness of these SE topics through surveys with students, professors and software professionals of universities and companies in Brazil. The partial results of this research suggest that some SE topics may be under-addressed insufficiently, while others may be less relevant to the undergraduate curriculum. Finally, we discuss about how apply less topics in order to form students with more competencies and abilities professional.**

*Index Terms*—**Software Engineering Topics, Teaching Methods, Preparing Graduates for Industry.**

## I. Introduction

The Software Engineering (SE) is one of the discipline of greatest importance in courses in the field of computing [1]. This arises from both the relevance of the software itself and the challenges related to the complete formation of a professional who will work in software industry. The result is an increase in demand by well-qualified software professionals [2]. Typically, these professionals learn in undergraduate courses as a way of preparing to the industry [3].

### *A. Identification of the Issue*

The software industry, specifically the Brazilian market, has shortage of suitably qualified professionals to work in professions that involve stages of the software development process [4]. The industry complains that undergraduate courses do not teach the necessary skills to students so they can start performing your job efficiently [5]. In this way, software companies have to complement the knowledge of recent graduates with training and have to provide skills related to software development process [6].

According to Lethbridge et al. [7], this deficiency in the formation of graduates in the SE area is the result of an inadequate education. There are several difficulties in the SE education during the undergraduate [8][9][10]: (i) very content taught in a short time; (ii) low motivation students have to study the SE theoretical concepts; (iii) difficulties in preparing students for professional practice within academic environment.

### *B. Statement of the Position*

We identify that the professional training needs of undergraduate and software industry are similar as well as difficulties encountered in this education process (excess content, short time, etc.). Then, we research as the industry performs the training of professionals.

In Brazil, many software companies have adopted quality models such as CMMI-DEV [11] and MR-MPS-SW [12]. Most of the companies achieves this through adoption of Software Process Improvement (SPI) programs implemented by consulting companies. During these consultations, the consultants have observed that spend much of time and financial resources in team training. This is because many professionals involved in SPI programs do not have adequate knowledge on particular SE topics [13].

Thus, the SPI consultants adopt various training strategies such as workshops, mentoring or coaching for technical transfer of the specific practices included in process areas (eg. Configuration Management) in order to develop the competencies and abilities necessary for the team.

Based on these observations and based on our professional experience, as SPI consultants and SE professors, we have the following position about the issue of Software Engineering Teaching in undergraduate courses:

***POSITION: If the Software Engineering discipline adopt training approaches of Software Process Improvement programs, then the preparation of students to the software industry will be more appropriate than the current teaching approaches.***

## II. Software Engineering Teaching

### A. Background Information

According to the ACM/IEEE [1], the SE is a discipline interested in the application of theory, knowledge and practice for the effective and efficient development of software systems that meet users' requirements. To fulfill the users' needs, SE professionals must have the ability to understand software development as a process and to ensure deadlines, costs, and quality of the product developed.

The *competence* term defines "the combination of knowledge, skills and personal attributes that are acquired through education, training and experience" [12]. Already the *ability* term defines "the individual's capability to do something, working in groups, learn to communicate and express themselves" [14].

The recommended approach to develop these abilities professional and competencies during the undergraduate is software development in capstone projects. This approach allows seeing the fruits of one's labor than when working alongside student teams as they "put it all together" and produce a real software product [15].

### B. Supporting Evidence

Wangenheim and Silva [5] performs a survey about the relevance of the topics covered in the Science Computer courses from the opinion of professionals in this area (specifically with regard to the SE). Their aim is to contribute to the improvement of current SE curricula.

These results indicates that there is a lack of attention to certain SE topics considered relevant. For some topics, we can identify a total lack of consideration. For example, the "Software Configuration Management" topic, which in practice is considered as an essential basis not only for software engineers, but also for any professional software.

In general, considering the suggestion of a total of at least 280 hours for a Computer Science course [14], the allocation of about 36 hours to SE topic does not to correspond with the perception of the importance of these topics.

There is literature on project-based learning within computing courses as a means to learn soft skills and complex technical competencies. It is propose a teaching approach to integrate contextualized project experiences with Software Engineering fundamental concepts. This approach, called the Software Enterprise, defines a delivery structure integrating established learning techniques around a project-based contextualized learning experience.

Gary et al. [15] propose a pedagogical model for the SE teaching in undergraduate courses in computing. This proposal, which was matured during 9 years of implementation at Arizona State University, combines traditional classroom with Problem Based Learning (PBL). Thus, the students attend lectures and put the concepts into practice through development lab sessions in each week. In this approach, the professor performs the role of coaching and the veteran students performs the role of mentoring.

The Gary et al. pedagogical model [15] is the main supporting evidence for our research, which add the use of SPI training practices (like mentoring, coaching, dynamics, etc.). For example, the specification of SE area roles (managers, developers and quality analysts) that aims to define a set of topics that students should learn to develop certain skills and abilities professional to work in the software industry.

## III. Discussion of the Issue

From the issue exposed in Section I and based on the background presented in Section II, we can highlight some observations:

i) Most of the software professionals are trained in computing in undergraduate courses [3];

ii) The software industry is unsatisfied about the level of preparation of undergraduates that entering in the software market [7];

iii) Bachelors that working as software professionals learn more about SE topics after the undergraduate course [5];

iv) There are several difficulties in the SE education during the undergraduate [16], like very content, short time, theoretical concepts.

### A. Questions Unanswered

In order to meet the objectives of this research and try to solve the issue, it is necessary to answer the follow questions still unanswered:

- **QU1.** ***What Software Engineering topics are addressed in undergraduate?***
- **QU2.** ***Is the learning of students in Software Engineering discipline related to the amount of the topics covered?***
- **QU3.** ***Are the students learn in a more appropriate way if we contextualize the application of Software Engineering topics with the roles that they will play in a development process?***

### B. Discussion Point

Several studies propose teaching approaches of SE, such as Bessa, Cunha and Furtado [6], Gary et al. [15] and Braga [16]. However, these approaches restrict the evaluation scope to the course content or to the process and the resulting product of the realization of a practical SE discipline. Thus, these approaches does not prepare adequately the students to work in the software industry.

For example, Bessa, Cunha and Furtado [6] proposes a simulation game of real environments to assist the SE learning. Despite the games effectively capacitate students, they also restrict the coverage of certain curricular topics. Thus, this approach can not properly explore the amount of SE topics, because would become too complex, losing its main characteristic of teaching/learning. Already Braga [16] proposes a multidisciplinary approach that sets guidelines for application of SE topics in various disciplines of Computer Science courses. However, this approach is limited to the curriculum topics application, without worrying about the development of the skills and abilities professional necessary to efficiently perform the activities of software projects.

Unlike these, our approach will apply the SE topics according to the roles (project manager, developer, quality analyst, etc.) that students will assume in the practical design of the SE discipline. This approach is similar to training strategy adopted by SPI consultants. Thus, it may will focus on professional formation of students, rather than focusing only on the evaluation of the process and product generated by them during the SE discipline.

Finally, Gary et al. [15] propose a pedagogical model for the SE teaching in undergraduate courses in computing. The teaching approach proposed in this research is strongly based on Gary et al. pedagogical model, as describe in Section II - B.

## IV. Possible Solutions

The main objective of this research is to propose an approach to support the teaching/learning of SE topics based on the adequacy of SPI training practices.

A draft of this approach was discussed by the authors based on their professional experience, as SPI consultants and SE professors, and based on the pedagogical proposal defined by Gary et al. [15]. This draft is shown in Fig. 1.

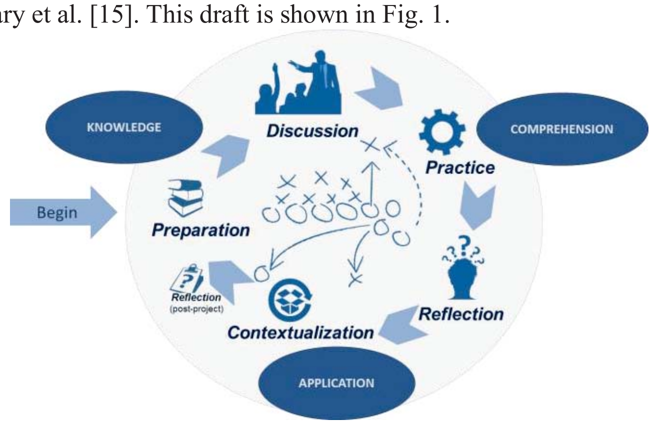



Fig. 1. Draft of the Teaching Approach

Initially, in the Preparation phase, students should perform reading and research of the SE topics. In the Discussion phase, seminars and debates related to these topics will held. Then, in the Practice phase the students apply the concepts learned through workshops of the using tools and through of group dynamics. In the Reflection phase, students record their expectations and hypotheses arise in relation to SE topics that will apply in practical project. The Contextualization phase will consist of the traditional software development approach by a group of students in order to contextualize the learning of these topics. At this stage, students will identify requirements with a real client and we will apply the mentoring techniques (through of professors) and coaching (through of senior students). Finally, students will carry out a new phase of Reflection, post-project, in order to compare the results achieved with your pre-project expectations and if their hypotheses were validated or refuted.

To define this approach, we need to perform some steps through research methods.

### *A. Literature Review*

To answer the QU1, we carried out a literature review in curriculum guidelines of the ACM/IEEE [1] and of the Brazilian Society of Computing (SBC) [14]. The aim of this literature review is to identify which SE topics are contemplated in these curriculums. In order to meet this aim, we carried out a mapping between these SE topics and the specific practices of the process areas of the CMMI-DEV [11] and MR-MPS-SW [12] models.

In addition, we carried out a systematic mapping [17] to identify the training strategies that the consultants apply in software industry in order to plan the adaptation of this strategies to academic context.

### *B. Academic and Industry Surveys*

Currently, we are conducting a survey (S-I) with professors who teaches SE disciplines in order to identify topics and teaching approaches are adopted. The results of this survey can support our position, finding that these professors do not address certain SE topics.

In order to respond the QU2, a survey (S-II) is carried out with students who have completed the SE discipline. The aim of this survey is to analyze the student learning and the teaching approaches that they consider more effective. The S-II results may support the H0 analysis, giving us evidence that certain SE topics may not have been effectively learned. In addition, we may correlate the S-I responses to the S-II responses in order to examine whether the teaching approaches adopted by professor are considered effective by the students.

Finally, a survey (S-III) is carried out with software professionals about which of their skills to perform SE activities were acquired during the undergraduate. In this survey, the aim is to find information about the relevance of the topics covered in the SE disciplines according to the opinion of professionals in this area. The results of S-III can reinforce our position, giving us evidence that the SE topics suggested in curricular guidelines meet (or not) the software industry demands reported by industry professionals.

The surveys are applied in public and private universities and companies in Brazil and follow the guidelines of Kitchenham and Pfleeger [18]. These surveys are released in e-mail list, SE area groups on social networks and in loco on public and private universities of the Belém and Recife cities. The surveys are available in http://goo.gl/vn5jHS and the survey protocol is available in http://goo.gl/gqzMrP.

### *C. Controlled Experiment*

In order to evaluate our proposed teaching approach, we intend to conduct a controlled experiment in a SE discipline in undergraduate in Computer Science. This experiment will follow the Wohlin et al. [19] guidelines, which will set the scenario for implementation of the teaching approach. First, the problem of a real client will be identified in order to define the requirements of software design.

Then, the professor will divide the class into two groups (experimental and control). The control group will follow the current teaching approach of the discipline, setting a traditional software development process (based on RUP, SCRUM, etc). The experimental group will adopt the teaching approach proposed in this paper, receiving specific training. Thus, this group may follow a new teaching/learning approach.

The purpose of this controlled experiment is to compare the students' learning of both groups through the analysis of concept maps (using the SE topics) produced by these at the beginning and end of the discipline. From this experiment application, we can to respond the QU3.

## V. CONCLUSIONS

In summary, the contributions of this research are: (i) the identification of SE topics in curriculum guidelines and the relevance analysis of these topics to the software industry; (ii) the identification, by means of surveys with professors and students of any problems in current SE teaching approaches; and (iii) an educational approach to meet, in a satisfactory manner, professional training demand for the SE area during the undergraduate.

Finally, we intend to publish the results to stimulate replication of this type of research and thus solve the gaps of this SE teaching area. In this context, we emphasize that the appropriate SE teaching is important to improve the current state of software development and help mitigate many of the traditional problems associated with the software industry.

As partial results, we obtained the answer from 46 undergraduate students and 23 SE discipline professors. These are from public and private universities from different regions of Brazil. Currently, we started the process of surveys data analysis. The partial results of this research suggest that some SE topics may be under-addressed insufficiently, while others may be less relevant to the undergraduate curriculum.

## ACKNOWLEDGMENT

The authors would like to thank Improvement Coordination of Postgraduate People (CAPES) for financial support to the development of this work.

## REFERENCES

[1] ACM/IEEE. Computer science curricula 2013. Curriculum guidelines for undergraduate degree programs in Computer Science. December 20, 2013. DOI: 10.1145/2534860.

[2] R. Duley, G. Hislop, T. Hilburn, and A. Sobel, "Engineering an introductory software engineering curriculum," in *Proceedings of the 16th Conference on Software Engineering Education and Training*, Madrid, Spain, 2003, pp.99-106.

[3] D. Nunes, C. Reis, and R. Reis, "Education in software engineering," in *The career of researcher in software engineering: principles, concepts and directions*, 1st ed. Salvador, Brazil: UFBA, 2010, ch. 5, pp. 132-181.

[4] ABES, "Brazilian software market: scenario and trends," in *Brazilian Software Market and Services 2014*, 1st ed. São Paulo, Brazil: Brazilian Association of Software Companies, 2014.

[5] C. Wangenheim, and D. Silva, "What software engineering knowledge is important for the professional software?," in *Proceedings of II Forum on Education in Software Engineering*, Fortaleza, Brazil, 2009.

[6] B. Bessa, M. Cunha, and F. Furtado, "ENGSOFT: Simulation tool for real environments to support the Problem Based Learning (PBL) in teaching Software Engineering," in *Proceedings of XX Workshop on Computer Education*, Curitiba, Brazil, 2012.

[7] T. Lethbridge, J. Diaz-Herrera, R. LeBlanc, and J. Thompson, "Improving software practice through education: Challenges and future trends," in *Proceedings of the Conference Future of Software Engineering*, Minneapolis, MN, 2007, pp.12-28.

[8] M. S. Soares, "An experience in teaching software engineering oriented practical work," in *Proceedings of I Workshop on Computer Education*, Vitória / Rio das Ostras. Brazil, 2004.

[9] J. Castro, I. Gimenes, and J. Maldonado, "A proposal for pedagogical plan for the Software Engineering discipline," in *Proceedings of II quality course of undergraduate of Computing and Informatics area*, Curitiba, Brazil, pp. 251-270.

[10] O. Hazzan, and Y. Dubinsky, "Teaching a software development methodology: The case of Extreme Programming," in *Proceedings of the 16th Conference on Software Engineering Education and Training*, Madrid, Spain, 2003, pp. 176-184.

[11] CMMI Product Team, "CMMI for Development, version 1.3," Carnegie Mellon University, Pittsburgh, Software Engineering Institute, Pennsylvania, Technical Report CMU/SEI-2010-TR-033, Nov. 2010.

[12] Association for Promotion of Brazilian Software Excellence, "SPI Reference Model for Software, version 2012," SOFTEX, General Guide SPI for Software, Dec. 2012.

[13] Leal, G. et al. 2012. Empirical study about the evaluation of the implantation of MPS.Br in enterprises of Paraná. In *Proceedings of XXXVIII Conferencia Latinoamericana en Informatica*. DOI = 10.1109/CLEI.2012.6427201.

[14] SBC. Reference curriculum for undergraduate courses in Bachelor in Computer Science and Computer Engineering, 2005.

[15] Gary, K. Lindquist, T. Bansal, S. and Ghazarian, A. 2013. A Project Spine for Software Engineering Curricular Design. In Proceedings of 26th Conference on Software Engineering Education and Training (CSEE&T). DOI = 10.1109/CSEET.2013.6595265.

[16] J. Braga, "Guidelines for the interdisciplinary teaching of Software Engineering," in *Proceedings of II Forum on Education in Software Engineering*, Fortaleza, Brazil, 2009.

[17] B. Kitchenham, and S. Charters, "Guidelines for Performing Systematic Literature Reviews version 2.3," Keele University, University of Durham, Keele/Durham, UK, Technical Report EBSE-2007-01. Jul. 2007.

[18] B. Kitchenham, and S. Pfleeger, "Personal Opinion Surveys," in *Guide to Advanced Empirical Software Engineering*, Springer, 2008, ch. 3, pp. 63-92.

[19] C. Wohlin et al., "Experimentation in software engineering: an introduction," in *Kluwer Academic Publishers*, Norwell, MA, 2000.

# SESSION

# ASSESSMENT STRATEGIES + STUDENT OUTCOMES + ISSUES RELATED TO ACCREDITATION

## Chair(s)

**TBA**

# An Eight-Level Tree Structure Implementing Hierarchies of Program Assessment Processes

Iraj Danesh
Department of Mathematics and Computer Science
Alabama State University
Montgomery, AL 36104
334-229-4355
idanesh@alasu.edu

## ABSTRACT
This paper, based on the principles of good practice for assessing student learning [4], develops and implements hierarchies of program assessment processes in an eight-level tree structure. It starts with program at the root (level-0), subdividing major assessment steps into particular measurable steps (top-down design), continues all the way through several levels of assessment processes and ends with level-7 (leaves) where tools of measurements are developed and administered. Bottom-up traversal of the tree completes progress of data consolidating, affecting improvements based on the analysis of the results.

## Categories and Subject Descriptors
k.3.2 [**Computers and Education**]: Computer and Information Science Education- *Accreditation, self-assessment*

## General Terms
Design, measurement

## Keywords
Assessment, outcomes, enabling to attain, continuous improvements, rubrics, collecting, sampling, mapping, consolidating, analyzing, documenting, reporting

## 1. INTRODUCTION
Accrediting agencies such as SACS, NCACS, and EASC grant status of public recognition to the programs that meet the agencies' standards, criteria, and requirements. ABET criteria [1] such as Program Educational Objectives (PEOs), Student Outcomes (SOs), Continuous Improvement (CI) as well as data collecting and consolidating strategies, organizational structures (committees), methods, means and tools of assessments including the role of categories of Blooms' Taxonomy domains in the development of high level rubrics are discussed as guidelines for program assessment, self-study and accreditation purposes.

## 2. STRUCTURE AND LEVELS OF ASSESSMENT TREE
Figure 1 is an exemplar succinct tree structure, developed on the assumption of having five PEOs, nine SOs (ABET SOs 'a' through 'i'), six performance indicators of student outcomes (PISOs) and three courses for each SO, ten course outcome indicators (COIs) per course, as well as necessary means and tools of assessment. It includes eight levels of assessment processes providing an overall view of the program assessment tree for improvement continuity and attainment of SOs. Each level is associated with processes that accumulate specific data, exchanging feedback between layers above and below it. The role of PEO, SO, and course CI cycles (depicted in figure 3) are realized in leve1s 1, 2, and 4.

### 2.1 PEO-Level (level 1)
"*PEOs are broad statements that describe what graduates are expected to attain within a few years after graduation*" *[2, 14]*.They serve as targets for career development *(*e.g. become *CEOs and entrepreneurs.)*. To maintain continuity and attainment of PEOs, based upon the input from program constituencies, ABET and ACM/IEEE-CS recommendations, PEOs are reviewed and changes are adopted periodically (known as *Slow-Cycle* with a frequency of six years) by the CSC curriculum Committee. Figure 4 provides an exemplar excerpt of involvement of various constituencies and consulting entities in PEO development. Evaluation of achievements and attainment of PEOs are no longer required by ABET. However PEOs should remain affirmed and consistent with institution mission and constituencies' needs.

### 2.2 SO-Level (Level 2)
"*SOs indicate the ability of students to use the knowledge they gained at the time of graduation*"[6] (e.g. students are able to analyze computing algorithms). Default ABET/CAC recommended SOs 'a' through 'i' (referred to as *"characteristics*") can be found

in reference [6]. They prepare graduates to attain PEOs and are affirmed, assessed, periodically reviewed (every three years – through CI *SO-Cycle*), and changes are adopted. CI processes demonstrate how students are enabled to attain all the characteristics. Spreading of SO data collection over alternative years (table 1) is acceptable and even desirable.

## 2.3 PISO-Level (level 3)

PISOs assess SOs and are similar to leading economic indicators [3]. Every ABET SO is broken down into six simple statements called PISOs (figure 1), that are measurable aspects that allow one to determine the extent to which the outcome is met. Enabling characteristics does not mean goals are necessarily met. Success targets for achievement of PISOs are established and measured at this level (e.g.: At least 80% of students demonstrated excellent or good performance, and at least 90% demonstrated acceptable - targets are not achieved if improvement is needed). PISOs may constitute dimension components of an analytic rubric [3] with up to five levels of performance (scale). To assure that different instructors at different times characterize student performance consistently, a holistic rubric associated with performance indicator may be developed. Holistic rubrics are best fitted for PISOs data collection bringing uniformity, consistency, and play a significant role in faculty bias scoring. A sample holistic rubric for PISOs can be found in reference [11].

## 2.4 Course-Level (Level 4)

This level provides *Fast-Level Cycle* with a frequency of one semester or a quarter for continuous course improvement strategies, examines role of courses (through which the relevant skills, knowledge, and behavior are acquired), and develops course outcomes (COs) and COIs. It provides course displays demonstrating curriculum properly enables and attains all the characteristics. It also provides evaluation suggesting method of course improvements and their implementations.

## 2.5 COI-Level (level 5)

COIs are established to measure COs (like PISOs measuring SOs), documenting the role of tests, projects, and assignments in assessment of PISOs, and SOs. They may be measured through levels of concept, topic or subject in courses and are ultimate instruments for concrete measurements of PISOs in classrooms.

## 2.6 Means-Level (level 6)

For each course, appropriate areas of measurement or means (tests, assignments, participant observation, oral and written presentations, group project, capstone project, etc.) are designed to assess COIs. Most projects rely heavily on team projects and in-class teamwork. Functioning within a team is in harmony with ABET SO “d”. Team coherence and techniques for achieving such coherence must be explicitly assessed using team rubrics.

## 2.7 Tool-Level (level 7)

Appropriate use of different tools (rubrics, faculty panel, item analysis, and percentiles, etc.), different types of rubrics (holistic or global, analytic, weighted, etc.), *components* of an analytic rubric (*dimension* or performance indicator, *scale* or level of performance, *descriptor* or expected result), including *attributes* of dimension (content referent, action verb, value free) generates actionable data for analysis and evaluation, affecting improvement and providing feedback. Variety of sample illustrative rubrics can be found in reference [7].

# 3. SELECTING AND SAMPLING

Good assessment demands good compromises. Using too many of every available instrument (assessment methods, rubrics, data, courses, and students), may generate extensive raw data with little information. Not all large multiple sections of courses, methodologies (with their own advantages, disadvantages, and caveats), means and tools can be used for a single assessment. Appropriate selection of instruments, limiting total number of courses to six (3 for each PO - e.g.), and sampling of students representing all students of all point averages avoids ambiguity, resolves caveats concerning methodologies and number of instruments, and eventually reduces the workload.

# 4. MAPPING AND ALIGNMENT

Exemplar excerpts of mapping and alignment among PEOs, SOs, PISOs, courses, COIs, assessment means and tools are provided in graphical and tabular forms (figures 1 & 2 – table3). Figure 2 provides Bottom-up traversal view of involved entities in the tree structure showing their order of precedence. Table 2 provides an exemplar excerpt of rubric topics that are mapped with course topics and means of assessment.

# 5. REPORTING AND IMPROVEMENT

EAMU succinct performance vector (PV) [8], and Four-Column Template [10] are favorite choices for both assessing and reporting (data presented here are for illustrative purposes only and are not actual). EAMU is the acronym for Excellent-Adequate-Minimal-Unsatisfactory. “EAMU” PV transforms data collected from direct assessment into succinct vectors of information. Table 4 shows “EAMU” PV table for an annual report of a PISO assessment for ABET SO ‘i’ (e.g.). “EAMU” PV for courses indicates that PISO is of concern. The expected success targets for courses I and II were met, but not for course III, implying content of course III needs to be modified and improved. The number of students in courses I, II, and III are 17, 12, and 9 respectively. Course I assessment results (e.g.) may be reported as: **EAMU vector (8, 1, 7, 1)**, meaning out of 17 students, there are 8 excellent, 1 adequate, 7 minimal and 1 unsatisfactory.

Based on the Nichols Five-Column Assessment Model [13], a modified Four-Column Template (table 5) is designed to

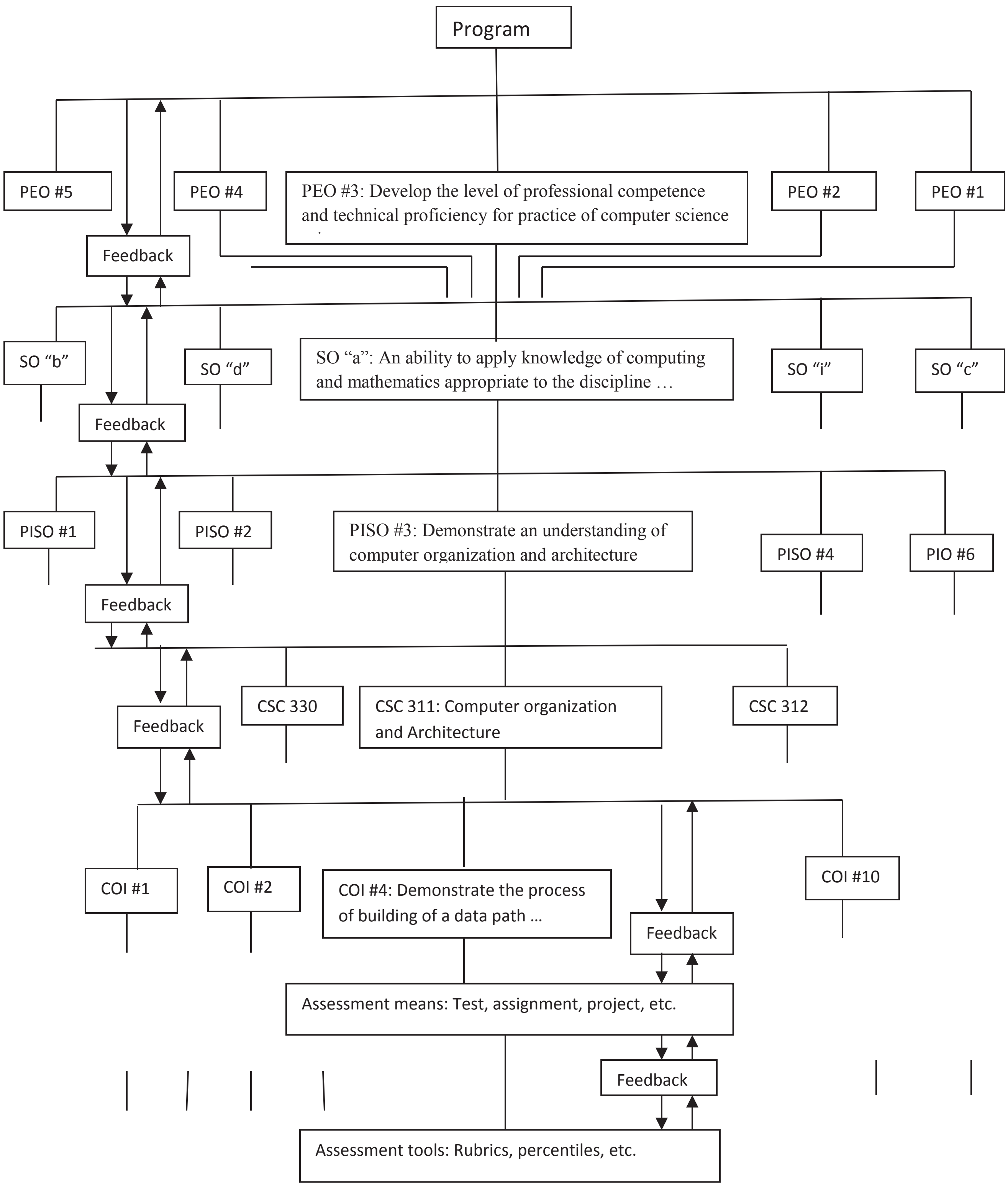



**Figure 1. An exemplar excerpt block diagram that provides mapping, alignment, feedback exchanges and hierarchies of program assessment in a tree structure**

**Table 1. Spreading of SO data collection over alternative years (every three years)**

| SOs | Year1 | Year2 | Year3 | Year4 | Year5 | Year6 |
|---|---|---|---|---|---|---|
| 'a' | Data collected | | | Data collected | | |
| 'b' | | Data collected | | | Data collected | |
| 'c' | | | Date collected | | | Data collected |
| 'd' | Data collected | | | Data collected | | |
| …. | | …… | | | …….. | |

**Table 2. An exemplar excerpt of rubric topics that are mapped with course topics and means**

| Tool (rubric) topic | Course topic | Means: Assignment, test, and project |
|---|---|---|
| Implementation in a high-level language | Manipulation of data structures, recursion, etc. | Test #1, question 5,6; programming assignments 2, 3 |
| ……. | …….. | ……… |
| Data representation and design of algorithms | Stacks, queues, linked lists, binary trees, etc. | Test #2, question 4, project 3 |

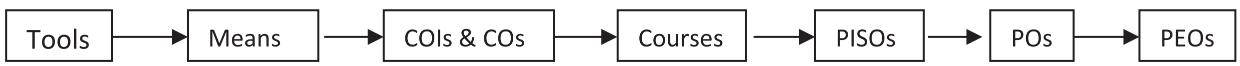



**Figure 2. A bottom-up traversal view of involved entities in tree structure for program assessment**

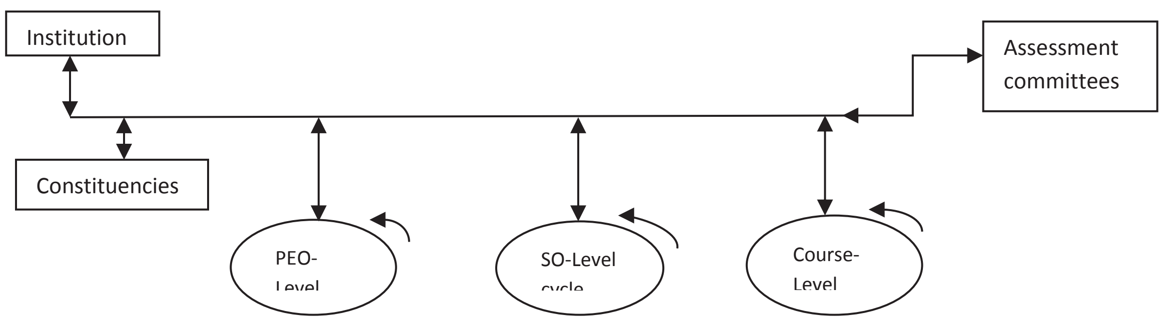



**Figure 3. An overall view of continuous improvement cycles in program assessment block diagram**

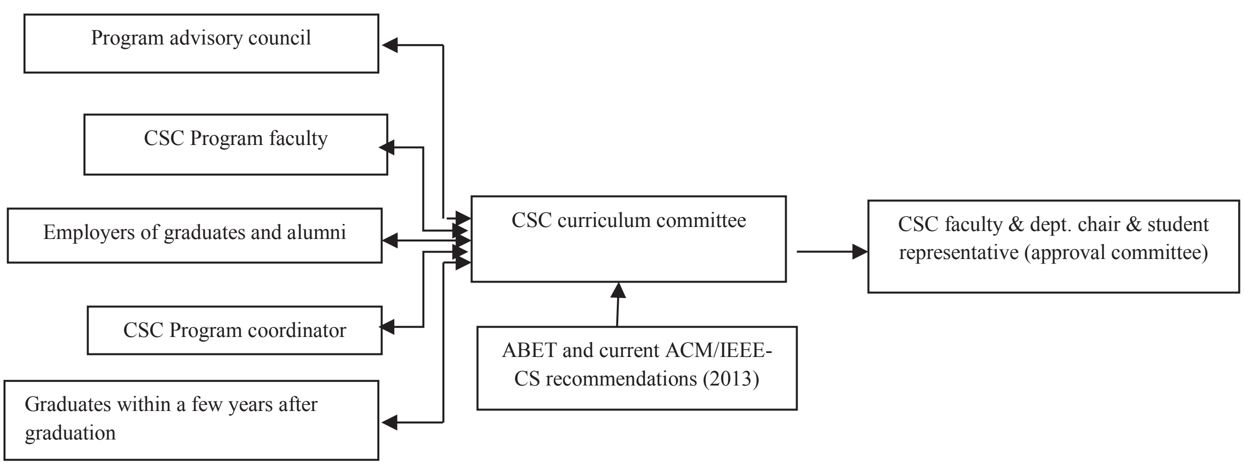



**Figure 4.An exemplar excerpt of involvement of various constituencies and counselling entities in developing PEOs**

**Table 3. An exemplar excerpt of PEOs that are mapped with SOs, PISOs and related courses**

| Program educational objective (PEOs) | Student outcomes (SOs) | Performance indicators of student outcomes (PISOs) | Content courses |
|---|---|---|---|
| 1-Develop the level of professional competence and technical proficiency for practice of computer science | (a) An ability to apply knowledge of computing and mathematics appropriate to the discipline | PISO #1:Demonstrate an understanding of computer organization and architecture<br>PISO #2: ……………….. ……………………..<br>PISO #6: Demonstrate an understanding of data structures and algorithm analysis | CSC 211, CSC 380, CSC 421, CSC 212, CSC 280, CSC 447 |
| | (b)An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution | ……………….. …………………….. ……………………. | ……………………. ………………….. ……………………… |
| Continue……… | Continue………… | Continue…….. | Continue……. |

**Table 5. An exemplar Four–Column template report of expected student outcomes**

| Expected Outcomes | Predefined success targets | Actual performance achieved | Affecting improvement/ action taken |
|---|---|---|---|
| ABET Student Outcome (a) *An ability to apply knowledge of computing and mathematics appropriate to the discipline* | 80% of students demonstrate excellent performance, and 90% demonstrate acceptable | 70% of students demonstrated excellent performance, and 75% demonstrated acceptable | Goal was not met. Students have difficulty with polymorphism that was addressed (e.g.) |
| ABET student outcome *(d)* *An ability to function effectively on teams to accomplish a common goal* | 80% of students demonstrate excellent performance, and 90% demonstrate acceptable | 90% of students demonstrated excellent performance, and 90% demonstrated acceptable | Goal was met. No action necessary. |

**Table 6. An exemplar excerpt of assessment report of PISOs for SO "a"**

| PISOs | Courses assessed | Unsatisfactory (head count) | Minimal (head count) | Adequate (head count) | Excellent (head count) | Total |
|---|---|---|---|---|---|---|
| PISO #1 | CSC 311 | 2 | 1 | 3 | 8 | 14 |
| ……… | ……… | …….. | ………. | ……. | …….. | ….. |
| PISO #6 | CSC 421 | 1 | 2 | 3 | 8 | 14 |

**Table 7. An exemplar excerpt of consolidated assessment report summary for ABET SOs**

| ABET student outcome | Below expectation (head count) | Meets expectations (head count) | Above expectations (head count) | Total # of students (head count) |
|---|---|---|---|---|
| SO "a" | 0 | 6 | 8 | 14 |
| ……. | ……….. | ……………. | ……… | …… |
| SO "i" | 9 | 4 | 1 | 14 |

**Table 4. A PISO PV table for ABET SO "i"**

| Name | U | M | A | E |
|---|---|---|---|---|
| I-Procedural Programming | 1 | 7 | 1 | 8 |
| II-Software Development | 1 | 0 | 1 | 10 |
| III-Object oriented programming | 1 | 0 | 5 | 3 |

incorporate processes of reporting as identified expected outcomes, predefined success targets, actual performance achieved, and affecting improvements (action taken) based on the analysis of the results.

## 6. CONSOLIDATE REPORTING

The results of PISO level is transferred into Table 6, providing an excerpt of head count rates of six PISOs for SO "a". For PISO #1, out of 14 students, there were 8, 3, 1, and 2 students in excellent, adequate, minimal, and unsatisfactory categories respectively. At SO level all information is consolidated in table 7, reporting cumulative head count rates for SOs annual report summary. It indicates that for SO "a" students are either meeting or exceeding expectations. For SO "i", nine students performed below expectations. Corrective measures for further improvement and sustainability were devised (addition of a unit on the solution of recurrence equations with expansion of recurrence relations).

## 7. ENABLING, ATTAINING, AND DOCUMENTING

ABET requires programs enable all graduates to attain all characteristics. Course displays including syllabi, exams, samples of student work (table 2), minutes of meetings, etc. demonstrate how curriculum enables all characteristics for all students. It is expected that mission of institution, PEOs, SOs be documented, published and visible to public (location includes web sites, catalog, etc.)

## 8. ASSESSMENT AND BLOOMS' TAXONOMY

Blooms' Taxonomy refers to a classification of the different objectives set for student learning by educators. It divides educational objectives into three "*domains*" (*affective, psychomotor, and cognitive*) [5]. Receiving, responding, valuing, organization, and characterization by a value are categories of affective domain. Perception, set, guided response, mechanism, complex or overt response, adaption, and origination are categories of Psychomotor. Knowledge, comprehension, application, analysis, synthesis, and evaluation are categories of cognitive domain [9, 12]. These categories may be used in the development of high level rubrics either as *dimension* (performance indicators) or as *scale* (level of performance). The higher the cognitive level, the more difficult it is to achieve targets. Thresholds and success targets might be lowered at high cognitive level.

## 9. CONCLUDING REMARKS

Simplicity favors regularity. Selection of small number of PISOs, appropriate methods and instruments, limited number of relevant courses and randomly sampled students (representing all), foster quicker improvements, and conforms to the philosophy of keeping assessment simple. The following agendas [4]:

- Learning the materials most valued to students and constituencies (educational values)
- Learning as multidimensional, integrated, and revealed in performance overtime,
- Keeping assessment continual and cumulative (not episodic)
- Meeting responsibility to students and stakeholders.

Serve as excellent vehicles for wider improvement and pedagogical enhancements. They partially constitute the fundamentals of good practice for assessing student outcomes (formerly program outcomes), and should be treated as such during assessment processes.

## 10. REFERENCES

[1] ABET (Accreditation Board of Engineering and Technology) Program Assessment Workshop, 2012 ABET Symposium, April 2012, St. Louis, MO.

[2] ABET 2011 definition of PEOs

[3] ABET Program Assessment Workshop, 2012 ABET Symposium, April 2012, St. Louis, MO.

[4] Astin Alexander W., Banta W. Trudy, "Principles of Good Practice for Assessing Student Learning" 210 ABET Faculty Workshop handbook, appendix B, page 2-3

[5] *Blooms' Taxonomy*, http://en.wikipedia.org/wiki/Blooms' Taxonomy, accessed on 7/4/213

[6] *Criteria for Accrediting Computing Programs*, 2012 – 2013, http:/www.abet.org/computing-criteria, 2012-2013, accessed on 7/11/2012

[7] Danesh Iraj "A General Course-Level Assessment Cycle for Computing Courses" Proceedings of the 2013 International Conference on Frontiers in Education: Computer Science and Computer Engineering, Las Vega, July 22-25, 2013, p. 48-54

[8] Estel John, *A Heuristic Approach to Assessing Student Outcomes Using Performance Vectors,* ABET Symposium, St. Louis, MO, April 19-21, 2012

[9] Gronlund N. E."*Measurement and Evaluation in Teaching"* New York: 4th ed., Macmillan Publishing, 1981

[10] Jones Lisa, "*Using the Four-Column Model to assess a Program"* ABET Symposium, St. Louis, MO, April 19-21, 2012

[11] Lakshmanan K. B. "*Assessing Student Learning in Computer Science – A Case Study*" Proceeding of the 2013 International Conference on Frontiers in Education: Computer Science and Computer Engineering, Las Vegas Nevada, July 22-25, 2013, p. 3-9

[12] McBeath, R. J."*Instruction and Evaluating in Higher Education: A Guidebook for Planning Learning Outcomes",* NJ: Englewood Cliffs Educational Technology, 1992

[13] Nichols James O, *"A Road Map for Improvement of Student Learning and Support Services Through Assessment,* New York: Agathon Press, 2005.

[14] Rogers Gloria, Faculty Workshop on sustainable Assessment processes, Annual ABET conference, Baltimore, Maryland, Oct 26, 2010.

# Improvement of Researcher Database System

**Tetsuya Oishi**[1], **Noriko Kuwano**[1], **Masao Mori**[2], **Eiichi Takata**[1], **and Shigeru Kusakabe**[3]
[1]Office of Institutional Research, Kyushu University, Fukuoka-shi, Fukuoka-ken, Japan
[2]Office of Contents Utilization and Institutional Research, Center of University Management,
Tokyo Institute of Technology, Meguro-ku, Tokyo-to, Japan
[3]Faculty of Information Science and Electrical Engineering, Kyushu University, Fukuoka-shi, Fukuoka-ken, Japan

**Abstract**—*Most universities around the world must be accredited in order to assure and enhance the quality of education. Japanese universities also are under accreditation by the government as well. They must undergo certified evaluation and accreditation periodically, which is conducted by organizations certified by the Japanese Minister of Education. Accreditation committees evaluate every university based on the Standards for Evaluation and Accreditation by reviewing the self-assessment reports. That is why it is important for universities to pursuit their advantages and to create high quality self-assessment report. To present some advantages in the report, the university staffs have to obtain the evidences demonstrating some effects or outcomes. Our university has the "Researcher Database System" in which all faculties can store their activities. This system is used for not only accumulating but also self-assessment. Moreover, the data which are accumulated in this system are made public in the "Kyushu University Academic Staff Educational and Research Activities Database". In this paper, we describe the outline of "Researcher Database System" and its improvement.*

**Keywords:** Database, Big data, Web System, Institutional Research

## 1. Introduction

There are two major evaluations for National University Corporation in Japan. One is Certified Evaluation and Accreditation, and another is National University Corporation Evaluation.

The former is the mandatory evaluation process for universities. This evaluation is covering overall conditions of education and research conducted by certified evaluation and accreditation organizations. In Japan, all higher education institutions are obliged to assess their activities in order to assure and enhance the quality of the institution. These institutions also have to undergo certified evaluation and accreditation periodically. This evaluation is conducted by organizations certified by the Minister of Education, Culture, Sports, Science and Technology (MEXT). Certified Evaluation and Accreditation was introduced to contribute to the further development of Japanese higher education in 2004.

The latter is performance-based evaluation of national university corporations in respect of their attainment of their mid-term objectives, mid-term plans and annual plans for education, research and management.

- *Mid-term objectives* mean individual corporate objectives that Japanese national university corporations should achieve over a six-year period. These objectives are made public.
- *Mid-term plans* are developed by Japanese national university corporations in order to achieve their mid-term objectives. These plans are made public too.
- *Annual plans* are developed by Japanese national university corporations in order to achieve their mid-term plans.

The National University Corporation Evaluation Committee is entirely responsible for this evaluation. This committee was established within MEXT pursuant to the Cabinet Order of October 1, 2003.

In order to obtain good evaluation, we have to explain our conditions of education and research obviously. Moreover we have to gather enough evidences to explain them. We have some web systems to gather evidences. There is the "Researcher Database of Kyushu University" which is one of these systems. The data which are inputted from this system are made public partially in the "Academic Staff Educational and Research Activities Database". We have improved the "Researcher Database of Kyushu University".

In Section 2, we describe the "Researcher Database of Kyushu University". Then we explain the improvement of the system in Section 3, and compare the condition before improvement with the condition after improvement in Section 4. Finally, we summarize our results in Section 5.

## 2. Researcher Database of Kyushu University

In Kyushu University, the "Researcher Database of Kyushu University" is operating. This is the system which is operating in our university on the web. All teachers in our university can enter various data into the system. The system is operated by a top-down management style. In brief, all teachers have to use the system. In this section, we explain this system.

## 2.1 Purpose

The "Researcher Database of Kyushu University" was created for self-assessment of all teachers in Kyushu University [1][2][3][4]. They can enter their educational and researching situations into the system. By entering various data into the system, the level of education and research for teachers is expected to be improved in our university. Then our university can achieve aims of our university and social missions.

The system is composed of data entered by teachers based on teachers' self-assessment. The major purposes of the system are as follows.

1) The data entered by teachers are sources to establish management policies or future plans.
2) Japanese national universities, including our university, must be conducted evaluation by third-party organizations. In order to report to the third-party organizations, the data entered by teachers are used as basic information or sources.
3) In our university, the teachers have to report their activities to our university every year. The data entered by using the system are also used to report them.
4) These data are made public. The activities of education, research, social cooperation, and international exchange are promoted by the public information. After we receive opinions and proposals against the public information from the world, we have to understand the requests from the world accurately.
5) We have to store the basic data related to educational and researching activities. We also have to survey these data.

## 2.2 Entering Data

All teacher in Kyushu University can and must enter their activities to the system. They have to enter their activities on their own responsibilities. Every department must decide whether the following descriptions should make public.

- Descriptions of defamation for other people
- Descriptions against facts
- Descriptions of injustices
- Descriptions against the aim of education and research

These departments, in which the teachers describe such things, have to urge the teachers to re-think these descriptions. All teachers have to make an effort to update descriptions they have already entered.

## 2.3 Data Application

We can take advantage of the data entered by teachers. We also have to obey the following things.

1) The data entered by teachers are never used except for the purposes mentioned in Section 2.1.
2) Some of the data, which are agreed to make public by the committee in our university, are made public in the "Academic Staff Educational and Research Activities Database" [5] operated by Kyushu University.
3) When someone would like to use the data based on the purposes shown in Section 2.1, he or she must be approved by the person in charge of the system.
4) When someone is approved of using the data, the fact must be reported to the committee.
5) The staffs who belong to Office of Institutional Research monitor the condition of using data.
6) When the data, which are made public in the "Academic Staff Educational and Research Activities Database", are provided to outsiders, the committee must approve it.
7) When making the data public or providing the data, we have to take care not to injure the specific individual right or profit.

## 2.4 Present Situation

In this section, we show the present situation on operating the "Researcher Database of Kyushu University".

### 2.4.1 Environment

This system is one of web systems and is working on a blade server. As we introduced a firewall, the network security is strong. The room where the server is placed is earthquake-proof. So the server is also protected physically.

This system is constructed by using Ruby. Ruby is one of object-oriented programming languages. Especially, we use Ruby on Rails which is an open source web application framework written in Ruby. We can customize the system freely because it is constructed by open source. The reason why we use the open source is that we have to develop the system. So, the system is working and is developing in parallel.

### 2.4.2 Restriction

All teachers can use this system on the Kyushu University network. But they can't use this system at the outside of Kyushu University. While we can enter the data of our presentations, papers, books, and so on, we can also enter the data of personal information such as students' names. The leak of personal information not only can cause bad reputation but also can lose confidences from the world.

We have to maintain this system. The way of maintenance is stricter than the way of using this system. In order to maintain this system, we can access this system using SSH port only from Office of Institutional Research, where we belong. We cooperate a certain programmer company. Even the programmers from the company can not access this system through SSH port except for our room.

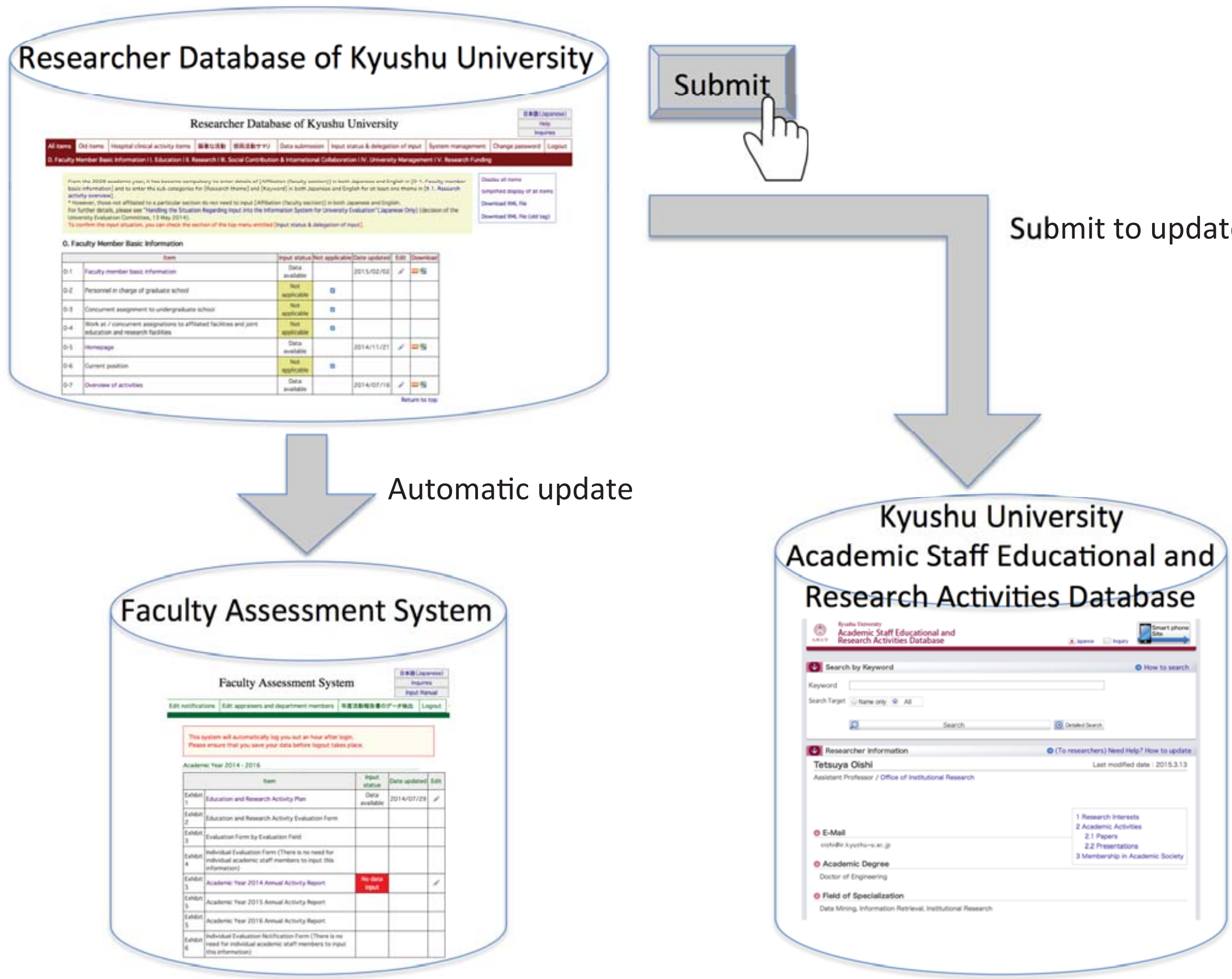



Fig. 1: The relation of three systems

We must protect the data the teachers entered. Moreover we must also prevent human errors. Then we introduced the very strong restriction.

### 2.4.3 Cooperation of Data to Outside

The data entered into this system are used in the "Faculty Assessment System" in Kyushu University. The data, for example, presentations, papers, books, and so on, can used in the "Faculty Assessment System". All teachers are assessed through "Faculty Assessment System" (Fig. 1).

We also have "Academic Staff Educational and Research Activities Database" system. This is the web system to make the teachers' information public. The teachers can decide whether they make each data public individually. Once they press the "submit" button, their information can be made public.

### 2.4.4 Cooperation of Data into Inside

We can use other data in order to enhance our data. In the system, we can enter the data of our papers. A paper data is composed of the title of paper, the authors' name, the authors' affiliations, the pages, and so on. It is very troublesome to enter these information. Then we use API provided by Thomson Reuter in order to enter these information automatically. At least, the bibliographic information is entered automatically.

However, the system can treat only text information such as the bibliographic information. Our university has own Library. The Library has repository system to store papers, books, and so on. We can relate the bibliographic information with its entity. This information of the entity not only is stored in the system but also is made public on "Academic Staff Educational and Research Activities Database".

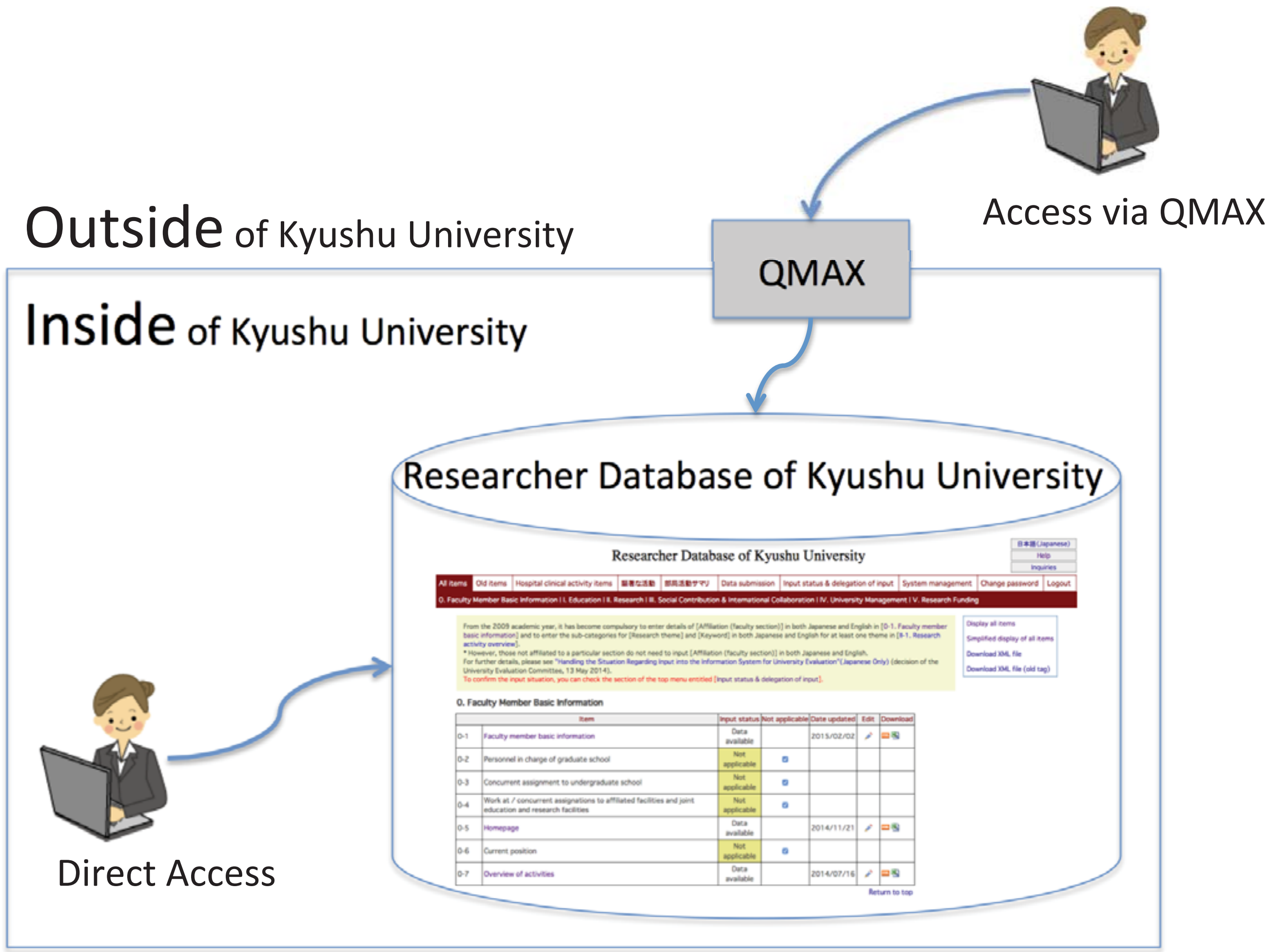



Fig. 2: Access to the system from inside or outside

#### 2.4.5 Input Situation

All teacher not only can but also must enter the data. We have check the situations how many teachers have completed to enter the data. There are two points of view when checking input situation.

1) A teacher has to enter all items into the system.
2) A teacher has to enter at least one data a year.

If breaking the rule, the department where such teachers belong is imposed a penalty. To say concretely, the subsidy to the department is reduced.

## 3. Improvement

Considering the present situation, we have improved the system many times. In this section, we show some major improvements.

### 3.1 Access from Outside

Basically, the system is accessed in Kyushu University only. However, many teachers thought that they would like to enter many data from outside (Fig. 2). As we have to prohibit all teachers from entering personal information from outside, we have permitted them entering a part of data except for personal information. What the teacher can enter from outside are "Presentations at academic society meetings, etc.", "Original papers", "Books authored", "Review papers, critiques, commentaries, book reviews, reports, etc.", and so on.

In order to realize entering from outside securely, we used QMAX [6]. QMAX is the omission of "Kyushu University Matrix AuthN System". QMAX is a reverse web proxy type SSO (Single Sign On) system. All teachers have own IC staff card that has matrix of numbers. They can login the system by using the number from the matrix.

### 3.2 Searching Papers

The teachers can insert the bibliographic information of papers into the system. For example, the bibliographic information are as follows.

- Paper title

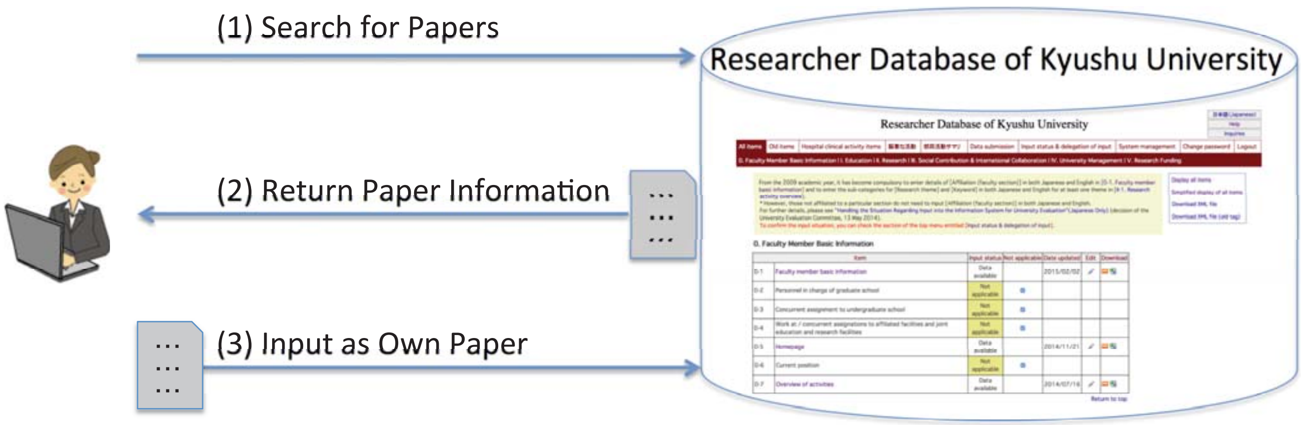



Fig. 3: Search for paper

- Authors' names
- Volume
- Number
- Starting page
- End page
- ISBN
- ISSN

The teacher can also insert other information except for the information listed above into the system, but it is very troublesome work. If there are the information we can use easily and automatically, we think that we can add information by using its information. So, as mentioned in Section 2.4.4, we decided to use the API provided by Thomson Reuter. The API can provide not only the major bibliographic information but also the other important information such as the number of citations.

If there are the information which have already been inserted into the system, we also think that we can use its information (Fig. 3). For example, we assume that a paper is written by two authors. If the first author have already inserted the bibliographic information of the paper into the system, it is useful for the second author to use the bibliographic information the first author wrote. In the system, the teachers can search papers which have been already stored and can insert bibliographic information into the system by using the information of the paper.

## 3.3 Authors' Names

All teacher can insert the basic information as a faculty member into the system. The teacher's name is included in the basic informations. The teacher can write his or her name freely, such as "Tetsuya Oishi", "Tetsuya OISHI", "T. Oishi", and so on.

The bibliographic information includes authors' names. In this system, the teacher can search other teacher as a author and include his or her name in the bibliographic information. In this case, the author's name to show is the string he or she has already inserted into the system. I give an example using my name.

I suppose that I have already registered my name as "Tetsuya Oishi". If the other teacher would like to register my name as an author, my name is shown in the teacher's information as "Tetsuya Oishi". If I have already registered my name as "Tetsuya OISHI", my name shown in other teacher's bibliography also becomes "Tetsuya OISHI".

I show an example which are shown in Table 1. When creating a bibliographic information by using the example, it becomes as follows.

> Tetsuya OISHI, Noriko Kuwano, Example of Title, Example of Journal, pp.51-58, 2015.7.

Then, I would like to show as follows.

> Tetsuya OISHI, Noriko KUWANO, Example of Title, Example of Journal, pp.51-58, 2015.7.

The differences of these appearance are the underlined words. In order to show the bibliographic information as the second appearance, I have to ask the second author to change her name to capital letters.

Therefore, we improve the system that the teachers can change the authors' name in the bibliographic information instead of changing the appearance of each name. The improvement can contribute the teachers in various fields as follows.

- "Tetsuya Oishi": Generally, the names are appeared on the bibliographic information by using the style which the first letters are the capital letters.
- "Tetsuya OISHI": In other fields, while the first letter is the capital letter in the first name, all letters are the capital letters in the last name.
- "T. Oishi": There are the teachers who would like to use the name style which has an initial of the first name and all letters of the last name.

All teachers became to think that the bibliographic information is beautiful by the improvement.

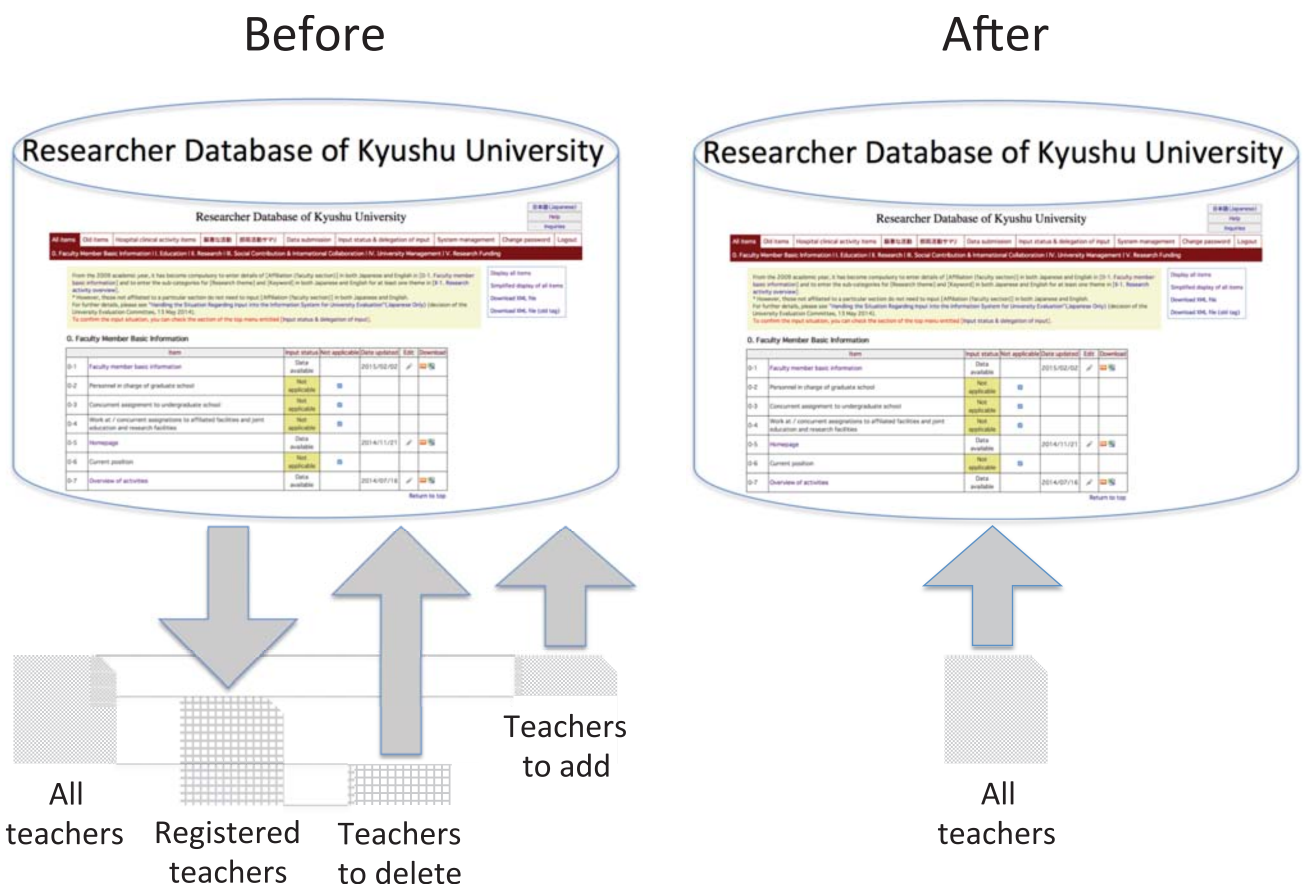



Fig. 4: Adding or Deleting Teachers

Table 1: A bibliographic information

| Name of item | Contents |
|---|---|
| Paper title | Example of Title |
| Author 1 | Tetsuya OISHI |
| Author 2 | Noriko Kuwano |
| Journal | Example of Journal |
| Starting page | 51 |
| End page | 58 |
| Year of publishing | 2015 |
| Month of publishing | 7 |

### 3.4 Adding or Deleting Teachers

In our university, while many teachers joined our university, many other teachers transferred the other university. Of course, while the former has to use the system, the latter has to become not to use the system. We also manage personal information per month.

Before improvement of the system, adding teachers was very troublesome work. We obtained the list of the teachers who belonged to our university. We compared the teachers who were included in the list with the teachers who were already registered in the system. While we deleted the teacher who were already registered in the system and who were not included in the list, we added the teacher who were not registered in the system and who were included in the list. We had to select the teachers to delete or add by using a relational database management system. We had to have long hours to do it.

Then we improved the system to delete or add teachers automatically. First, we insert the list of the teachers, who belonged to our university, into the system. The list that is inserted into the system now is compared with the list that was inserted into the system last month. In short, it means that the system became to be able to automatically do the task we did manually. We have to have only a half hour to do it. Actually, after shortening the time to do the task that we add teachers, we can provide the account of the teacher immediately.

## 4. Comparison of before and after improvement

As mentioned above, we have improved many functions. In this section, we compare before and after improvement.

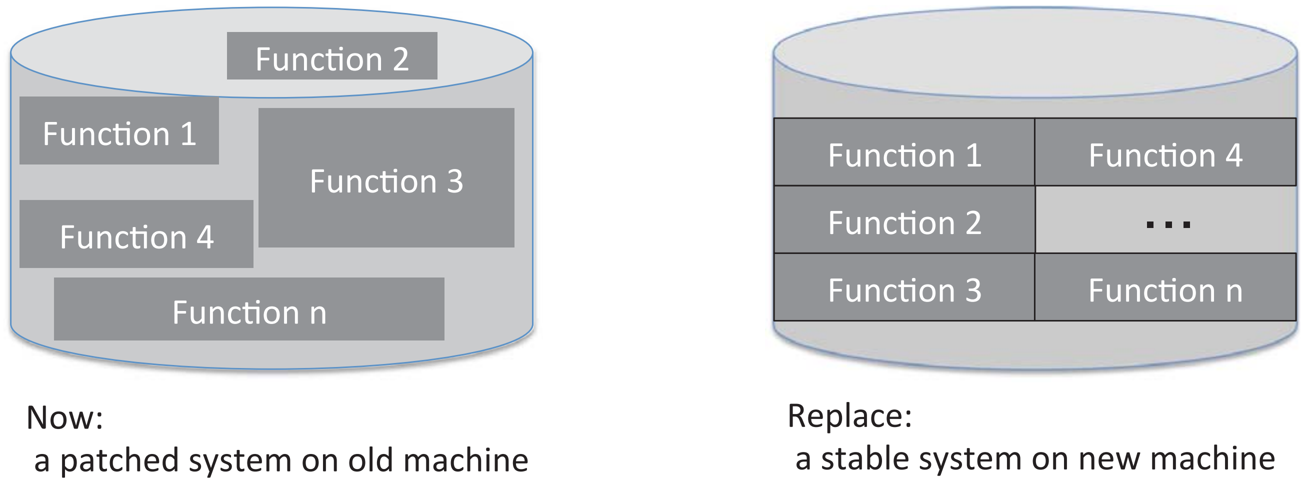



Fig. 5: Image of replace

### 4.1 Before Improvement

We had to deal with many complaints before improvement. We show the examples of complaints.

- The teachers who make many business trips would like to use the system from the destination. But they couldn't do it before improvement.
- There are too much items to be able to insert into the system. Especially, it is troublesome work to enter the information of papers.
- It is convenient to search authors from teachers in our university, but it is inconvenient not to write the authors' name freely.
- The teachers, who work in our university newly, would like to use the system immediately.

We had many trouble to solve, but we have already improved these functions as mentioned in Section 3.

### 4.2 After Improvement

As we have solved many problems, there become to be almost no complaints. Moreover, the duty to manage teachers' information is simplified. We think that almost all major problems are solved.

## 5. Summary

We have improved many functions in the Researcher Database System. Of course, there are minor improvements which we don't mention unfortunately. Until now, the system was in the developing phase. From now on, we have to operate the system certainly.

Then, we decided to replace the system with new one (Fig. 5). Actually, the machine, in which the system is working, is very old. Moreover, the system was created by patching many functions. Now, all functions are fixed.

The most important merit of replacing the system is that all teachers can use the system comfortably. Then all teachers can make their educational and research information public more easily. In 2016, the system will have replaced. We will make efforts to replace the system.

## References

[1] Masao Mori, Toshie Tanaka, and Sachio Hirokawa, "A Document Authoring System for Credible Enterprise Reporting with Data Analysis from Data Warehouse", *SEMAPRO 2010, The Fourth International Conference on Advances in Semantic Processing*, Oct. 2010.

[2] Kensuke Baba, Masao Mori, and Eisuke Ito, "A Synergistic System of Institutional Repository and Researcher Database", *Proceedings of The Second International Conference on Advanced Service Computing*, Nov. 2011.

[3] Masao Mori, Toshie Tanaka, and Sachio Hirokawa, "A Progressive Data Warehouse of Institutional Research with Web API and Mashup Visualization", *Third International Conference on Computer Supported Education*, Sep. 2011.

[4] Masao Mori, Toshie Tanaka, and Kensuke Baba, "Connecting an Institutional Repository with a Researcher Database", *International Conference on Advanced Applied Informatics - IRIM*, Sep. 2012.

[5] Academic Staff Educational and Research Activities Database. [Online]. Available: http://hyoka.ofc.kyushu-u.ac.jp/search/index_e.html

[6] For Simple & Safety Information Systems, Single Sign-On Web Portal. [Online]. http://web.sso.kyushu-u.ac.jp/English

# Assessing Student Outcomes with Comprehensive Examinations: A Case Study

**Jacob P. Somervell**
Department of Mathematics & Computer Science,
The University of Virginia's College at Wise, Wise, VA, USA

**Abstract**—*Among the many techniques for performing assessment for Student Outcomes, comprehensive examinations (sometimes called "exit exams") are an excellent choice. Highly adaptable, easily manageable, and directly related to assessment; comprehensive examinations should be considered by any department concerned with assessment of their programs. This paper presents a case study describing the implementation, use, and applicability of comprehensive examinations to one department's ABET assessment process. Other institutions could benefit from the lessons learned in utilizing this particular assessment method, and may be able to create their own assessment tools from this advice.*

**Keywords:** ABET, student outcomes, assessment, comprehensive examinations, accreditation

## 1. Introduction

Accreditation is (or is becoming) one the most important goals for graduate and undergraduate computer science, technology, and engineering programs across the United States and the globe. ABET is an international accreditation board that periodically reviews programs and awards accreditation based on rigorous requirements (see www.abet.org for details). The arguments for accreditation are many and beyond the scope of this paper, and the methods referred to are directly related to ABET assessment as part of their accreditation requirements. However, this does not preclude utilizing these techniques for other forms of assessment (such as local or regional assessment).

Part of the ABET accreditation requirements include student outcome assessment (criterion 3 and 4 [1], [2]). Student outcome assessment refers to one or more periodic assessment methods that determine the level of attainment of student outcomes [1], [2]. There are many ways in which programs can assess student outcomes, including both direct and indirect methods. While each category has strengths and weaknesses, in general, direct assessment methods provide more pointed feedback when issues arise. Comprehensive examinations are an example of a direct assessment – they utilize student work or student-produced artifacts to gauge the level of attainment of the student outcomes.

This paper relates the experiences one department had with using comprehensive examinations as a large part of their student outcomes assessment process for both Computer Science and Software Engineering programs. First a little background on assessment in general, followed by a description of the process used, a discussion of alternatives, a discussion of utility and success, and ending with challenges and thoughts. Part of the goal of this paper is to provide information on this technique for other departments who may need to create and/or update their assessment processes. *NOTE: This paper will deal exclusively with the Computing Accreditation Commission (CAC) and the Engineering Accreditation Commission (EAC) of ABET. The Applied Science Accreditation Commission (ASAC) and Engineering Technology Accreditation Commission (ETAC) may have slightly different requirements, but the overall assessment strategies should be applicable.*

## 2. Background on Assessment

Assessment of student work to understand how well students are meeting specified outcomes is part of the ABET accreditation process. A program seeking accreditation through ABET must have a periodic assessment process that determines the level of attainment of student outcomes [2], [1]. Student outcomes are defined by the program but must cover the required elements prescribed by ABET – the infamous "a through i" (for CAC) or "a through k" (for EAC).

After establishing its Student Outcomes, a program must then devise one or more mechanisms for measuring how well the program's graduates meet those outcomes. This collection of mechanisms is part of the program's assessment process. The design and utilization of these mechanisms is the source of the work and effort put forth by the faculty responsible for the program, hence, it is worthwhile to investigate which mechanisms provide useful information while minimizing faculty effort.

Among the many alternatives, a comprehensive examination shows promise as an assessment tool because it provides direct, usable feedback to the program with minimal effort by the faculty. Direct assessment methods utilize student produced artifacts to gauge the level of attainment of the program's outcomes.[3] If a problem is found through a direct assessment, often the nature of the assessment method indicates how to correct the issue. With indirect assessment, pinpointing how to correct an issue can be more problematic. Opting to use a comprehensive examination, developed by the faculty of a program, can provide direct, usable information in making changes to correct problems. The remainder of this paper details how one department leverages comprehensive examinations to drive their assessment.

## 3. Our Approach

Comprehensive exams are given each spring to the graduating seniors in both Computer Science and Software Engineering. These exams are created by the faculty and include myriad types of questions covering topics from core courses. These courses are required of all students moving through the degree programs and make up the core of the degree. In addition, the questions are designed to probe student knowledge and capabilities tied to the student outcomes.

The exam is reviewed annually by the faculty to ensure coverage of the core courses and to update any questions with current technologies and techniques. This allows the faculty to have complete control over the content of the exam. In departments that have assessment coordinators, those persons should also review the instrument with the faculty, but this step is not required. Figure 1 depicts the development process for the exam used by this department.

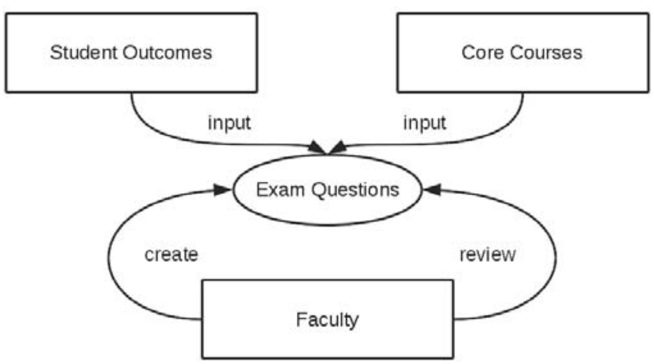



Fig. 1: Example of process used to develop exam questions. Note that the exam is directly related to the student outcomes.

After review, the exam is administered to the graduating seniors. There are many ways to achieve this. This department uses a 3-hour time block on a Saturday, late in the Spring semester (usually April). Paper versions of the test are used. Other possibilities include online delivery, convocation blocks (a dedicated block of time during which no classes are scheduled), evenings, or during regular class time. Each of these delivery methods has advantages and disadvantages relating to exam format, faculty time commitments, student time commitments, physical space constraints (finding a usable space on campus), and simple resource constraints (printing, copying, etc – for paper versions). Table 1 shows a breakdown of the relative strengths and weaknesses of each. This department is trying to get the exam into an online format but that process is slow.

Table 1: Trade offs for exam delivery options.

| | Faculty Time | Student Time | Resources | Flexibility |
|---|---|---|---|---|
| Dedicated Day | high | high | high | high |
| In Class | minimal | minimal | high | low |
| Convocation | moderate | moderate | high | low |
| Online* | minimal | minimal | minimal | high |

*after development of tool, development costs are high

The exam is split into three sections. The first section consists of 25 multiple choice questions and students are given 30 minutes to complete it. Question topics include core concepts from architecture, discrete mathematics, statistics, programming, operating systems, and others.

The second section consists of short-answer style questions. Students are asked to solve various types of problems from algorithms, database systems, networking, operating systems, programming language concepts, etc. Students are given one hour to complete about 20 questions.

The final section is discussion/programming. Students are given 90 minutes to complete 8 in-depth questions probing topics like ethics, program design, implementation, and selections from elective courses. Not all students take the same electives so students are allowed to choose any two questions from a list of sixteen. These questions are often updated in the yearly review to cover recent offerings of the elective courses.

Other test formats are possible. Instead of utilizing multiple choice and short answer, the entire test could be made up of discussion questions and more intense programming activities. A format like this might require lab space to ensure all students are on equal footing and are not cheating. Clearly other combinations are viable and each department can decide the best format for their situations.

## 4. Impacts on Assessment

Those programs seeking ABET accreditation must have some process for assessing their student outcomes. Many institutions utilize student work from course assignments as part of their direct assessment methodology, but course grades are typically not used [4]. Using student work is in fact an excellent method in that it provides directed feedback about student outcomes. However, using course work for assessment is very time consuming for the faculty involved and for any outside assessment coordinator. There are automated mechanisms that can reduce effort and facilitate both analysis and evaluation [5], [6], [7]. It is no secret that assessment is time consuming and faculty seek methods that require minimum effort with maximum benefit. Course based assessment is often simply too time consuming for the faculty.

## 4.1 Time Commitment

Comprehensive exams can require less time from the faculty. Take our department's approach as an example. The initial creation process is certainly time consuming, but once created, review by the faculty can take under 1 hour of time per faculty member, as many of the questions can be reused from previous offerings. Using in-class assessment techniques can take 1 hour per outcome just to collect the data and perform the assessment (does not include evaluation). After the graduates complete the exam, it must be graded. This is usually done in parallel with the exam (recall the exam is given in sections), thus grading can begin on the first section while students are taking the second, and so on. Overall time commitment from the faculty grading the exams is about 2 hours additional, for a total of 5 hours invested by the faculty (usually two faculty members) on the test day.

Another consideration is the time faculty must devote to the assessment evaluation and analysis. Using student work samples from an existing class requires extra time from the faculty to fill out whatever rubrics might be used and submit the analysis for evaluation. This is either done by the faculty or by dedicated assessment personnel. In contrast, the evaluation and analysis for the comprehensive exam is typically done by one faculty member and can be completed in about 1 hour (after grading is complete).

Modifying the assessment instruments is also a time consuming process. With traditional methods (utilizing student work samples), changes to the assessment process must be vetted and approved through the department. With a comprehensive examination, questions can me modified/removed/added during the review phase with little need for a lengthy approval process. Furthermore, these types of edits to a comprehensive exam are usually small changes/modifications to align questions to course content. The experience in this department is that review/editing process only takes about a day for complete turnaround. Most likely this is due to the fact that when there is a "good" question, it rarely needs to change.

Another source of time requirements for faculty involves the recording of the actual results and performing the requisite analysis and evaluation against the student outcomes. A comprehensive exam is comparable to traditional direct assessment methods but some speedup can be gained due to the mostly static structure of the exam and the fact that the questions are mostly the same across different versions. This structure also provides an excellent longitudinal analysis that may or may not be viable with traditional assessment. The next section details how the comprehensive exam is used for evaluation of student outcomes and how changes can be made to the program to address concerns found through assessment.

## 4.2 Analysis and Evaluation

After the exam is graded, analysis and evaluation can commence. This department uses program coordinators (a faculty member) to take the graded exams and perform the student outcome assessment. When implemented correctly, the comprehensive exam questions are tied to specific outcomes and the student performance on those questions provides the level of attainment of that outcome. Hence, the analysis process is very quick.

Evaluation includes determining how well students are attaining the outcomes and in situations where target attainment is not met, determining various courses of action to correct the problems. Because the test questions are tied directly to the student outcome, and because the questions cover specific course materials, it is straightforward to determine any corrective actions.

For example, suppose that a program has a student outcome dealing with "current technical skills". Furthermore, suppose that the student performance on the exam shows weakness in this area. [This would be known by looking at the questions tied to that outcome and the student performance on those questions.] In this case, suppose it is found that all students missed two questions related to Operating Systems concepts dealing with memory management. It would be up to the department to decide if they needed to stress that topic in subsequent offering of their Operating Systems course. If so, that would be an example of a program change that resulted from assessment.

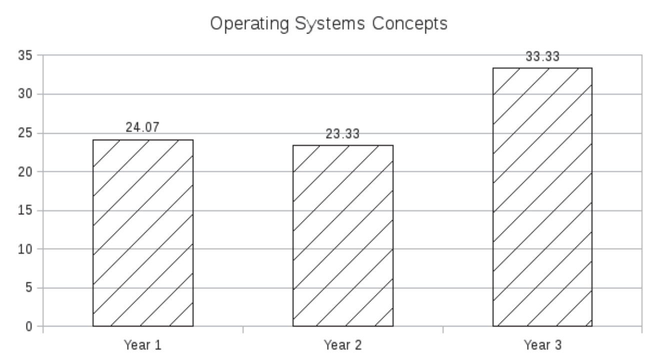



Fig. 2: Sample of Operating System Concept performance over three years. Percentage is percentage of students who answered correctly.

A more likely scenario would be for the department to note this problem and look at it again in the next cycle (the next year for this department). Because the test is designed to use similar (if not the same) questions from year to year; this type of analysis can be performed longitudinally by comparing current graduates' performance to previous graduates' performance and looking for trends. Figure 2 illustrates Operating System Concepts performance for three years. At this point, the department might decide to make a change concerning the Operating Systems course. An example change might be to include more assignments on memory management (paging and/or segmentation), extra time during lectures, or extra reading assignments (the nature and extent of the change would be up to the faculty of the department). Following the change, the department should then monitor that performance to determine if the change was successful, in other words, did the change in the program result in improved performance on the Operating Systems concepts questions on the comprehensive exam? Answering that question might even take two or three assessment cycles before the impacts can be observed.

To further illustrate the point, consider figure 3. This chart shows the student performance on the same concepts as figure 2, but after a program change. Here you can see that the change made in year 3 seems to have improved things in year 4, 5, and 6. Program “improvements” do not always result in improvements. Frequently the attempted change by the program ends up having little or no impact on the student performance indicators, so the department needs to keep close track of issues and follow up periodically on any changes made to a program.

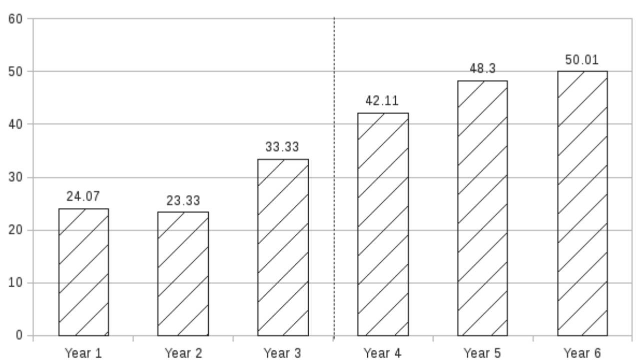



Fig. 3: Percentage of students who answered the Operating Systems questions correctly. Notice the break between year 3 and year 4. Student performance increased after the program change to address the concern.

With this form of assessment, in situations where a change is warranted, the department has documented evidence to support the change. Also, subsequent analysis can readily show if the changes resulted in student improvement, thus completing the assessment loop.

## 4.3 Flexibility

One desirable quality of comprehensive examinations is flexibility. Course based assessment methods are directly tied to the course and the assignment for which they are intended. Modifying these methods can be costly. Consider what happens when the instructor for a class leaves, or a new instructor takes over a course. That person will now be responsible for the assessment in that course. This implies certain training for how the new faculty member on how to perform the assessment; and the new faculty member must deal with the prescriptive nature of the assessment utilizing course assignments. That instructor must use the same assignment (or one very similar) in order to attain the assessment data and this prescribed work may be in conflict with that instructor’s teaching style or course management.

Contrast that with a comprehensive exam. Instructor turnover is no longer an issue because the same (or similar) questions can be used in every assessment cycle, regardless of the instruction technique or nature of the course. The only real requirement is that the content of the course be similar from offering to offering. This requirement is pervasive, that is all courses have this requirement, whether they are used for assessment or not. This allows for flexibility in teaching, assignments, grading, etc. Course-based assessment restricts this flexibility.

Another area of flexibility is the ease of adding questions to deal with new or updated student outcomes. As a program’s assessment process matures over time, often new outcomes (or modified outcomes) appear. With traditional course-based assessment, this would require modification of existing courses (or the creation of new courses) to include assessment of these new outcomes. With a comprehensive examination, all you would need is to create a new question (or set of questions) to assess the level of attainment for the new outcome.

For example consider the following possible student outcome:

*graduates will understand the need for life-long learning.*

This outcome is related to ABET student outcome ‘i’ [1], [2]. How could this outcome be assessed? With traditional course-based assessment this could be difficult. Specific assignments from some course would need to address this topic and then some student produced artifact would need to be evaluated and scored against some target. It is not clear how this should be done. Contrast that with a comprehensive exam question:

*Explain why technology majors need to update their skill sets. Give examples of ways a person could stay up to date in his/her field.*

A valid answer should include information about the ever-changing nature of technology and the rapid development and adoption of new techniques in related fields. Specific mechanisms for staying up to date might include certificates, pursuing advanced degrees, taking courses, online tutorials, membership in professional organizations, and many others. Students who correctly answer clearly have an understanding of the need for and the ability to engage in life-long learning. Furthermore, it would be obvious if the students had issues with this outcome. The ease of adoption and inclusion in a comprehensive exam far outweighs the cost of including this type of assessment in some course.

Even student outcomes that might initially seem difficult to measure with a test question can still be assessed using a comprehensive exam. Consider ABET outcomes d (teamwork), f (ethics), and h (societal impacts) [1], [2]. Each of these might seem like poor candidates for test questions, for example how can you gauge teamwork on a test? One way would be to use scenarios or cases and have the students provide solutions to hypothetical difficulties. The following question could be utilized on a comprehensive exam to gauge the students' understanding of teamwork:

*You and two others are assigned a group project. One of your team members fails to come to group meetings and has not been contributing any code nor giving feedback on written reports. It is still early in the project cycle and you need to deal with this issue. What actions would you take and why?*

Depending on the answers provided, it should be clear which students understand teamwork and its management and which students might struggle with the concepts. Granted, there may be better mechanisms for assessing these types of outcomes, but using cases and scenarios can give a department enough information to perform assessment.

More radical methods might be to include teamwork sections on the test. That is, allow pairs or groups of $n$ students to work together on parts of the test. Have the students then fill out teamwork assessment individually as part of the exam. This technique would certainly have challenges but is a viable alternative.

With some thought and creativity, each of the aforementioned "difficult" outcomes could be assessed using a comprehensive exam approach; thus allowing for complete, comprehensive assessment using one mechanism.

### 4.4 Comparison to Standardized Tests

It should be noted that standardized comprehensive examinations exist for Computer Science. The Major Field Achievement Test (MFAT) [8] is one such example. These tests are similar to other standardized tests like the SAT [9] and GRE [10]. The tests are designed to cover Computer Science topics utilized by most undergraduate Computer Science programs. These tests even include subject area performance indicators for common topics like Operating Systems, Architecture, and Discrete Mathematics [8]. The MFAT could be a useful tool for limited assessment because the results are aggregated and provide more of an indirect assessment (copies of the test questions tied to the students' responses are not available for analysis). Contrasted with the many benefits of tailored comprehensive examinations, standardized tests simply cannot provide the level of detail required for adequate outcomes assessment.

### 4.5 Other Benefits

There are other reasons that make comprehensive examinations desirable. Many programs provide mechanisms for transfers and degree completion. These types of students have variation in their backgrounds and often enter a degree program at varying skill levels when compared to traditional students. At the course level, some of these students might transfer sophomore, junior, and sometimes senior level content. If a department relies on in-class work for assessment, these students will literally be un-assessable. For very large programs that use various statistical sampling methods, perhaps it is not an issue; but for degrees with smaller numbers of graduates, each graduate contributes to the assessment (or should be included). A comprehensive examination would "catch" all graduates and allow for more thorough assessment using all students who proceed through the degree.

## 5. Lessons Learned

There are several important takeaways from utilizing comprehensive examinations for assessment:

- The flexibility of the comprehensive exam (including format, delivery, content, and malleability) is a huge plus for assessment.
- The potential for longitudinal analysis is truly useful for finding weaknesses, making appropriate changes, monitoring the results, and documenting the entire process (closing the loop).
- The overall time commitment by the faculty is not onerous. Initial creation of an exam will be the most significant time sink. Subsequent review, administration, and grading would be yearly time investments. Analysis and evaluation would again be a time commitment but overall these time requirements are localized, often to a single 5-hour day for a single faculty member.
- Online administration of comprehensive exams is a desirable goal. The time savings for faculty can be maximized without sacrificing the utility of a locally produced and maintained examination.
- Use of comprehensive examinations for the majority of assessment is not common. The most common method is to use course-based approaches, leveraging student work samples from assignments tied to student outcomes. The uniqueness of using comprehensive exams can be strange for visiting evaluation teams and care must be taken to illustrate the process and communicate its usage. This department has successfully secured both EAC and CAC accreditation while utilizing comprehensive exams as the majority part of its student outcome assessment processes – so it can be done.
- One potential pitfall with the use of comprehensive exams would be the motivation for students to perform at their best. Traditional course-based assessment

techniques have built-in motivation – students likely will do their best work on graded classwork (*With the tacit assumption that students care about their grades*). Without any kind of incentive, what reason would a student have to perform well on a comprehensive examination? Furthermore, why would he/she even show up? Many incentives exist, but the most powerful is to make the exam part of a required course. Students will then treat the exam like other course work and presumably perform at their highest level. Other motivations include awards (school wide recognition), prizes (tablets, music players, etc), and money.

- This department has found that assessing outcome 'g', which deals with communication [1], [2], is actually better through a capstone experience. That is, finding an exam question that allows for assessing a student's oral and written communication skills is extremely difficult to create. Granted, written communication can be assessed through the discussion type questions that already exist on the exam; but oral communication is different. To adequately assess this outcome, one must evaluate the student's ability to speak and communicate information through presentations (oral). It is possible to have an oral part of the examination, whereby students "defend" their work (similar to a thesis defense), but that technique has not been employed by this department. Instead, the required capstone experience is used for assessing oral communication skills (all students must present their capstone project work to the department faculty). There may be alternative approaches that could be included in the examination process, and that would allow for complete assessment through a single mechanism (the exam).

## 6. Conclusions

This paper has presented the experiences one department has had from using a comprehensive examination as the basis for student outcomes assessment for several years. These exams can be quite useful in support of student outcomes assessment. They provide flexibility in outcome coverage, excellent analysis, clear guidance when corrections are needed, easily documented evidence of "closing the loop" on assessment, minimal time requirements, and overall adaptability for almost any assessment need.

For programs considering ABET accreditation for the first time, or for programs that undergo periodic assessment review from local and regional accreditation boards, finding a process that allows for easy assessment, analysis, evaluation, and documentation is somewhat of a holy grail. The unfortunate reality is that no such systems exist yet. However, comprehensive exams can provide compelling benefits when compared to other techniques and might be a good first step for many programs.

## References

[1] ABET. Eac criteria for accrediting engineering programs, 2015-2016. [Online]. Available: http://www.abet.org/eac-criteria-2015-2016

[2] ——. Cac criteria for accrediting computing programs, 2015-2016. [Online]. Available: http://www.abet.org/cac-criteria-2015-2016

[3] A. Koohang and T. Smith, "Direct assessment of student outcomes: A systematic approach," *Issues in Information Systems*, vol. 13, pp. 1–10, 2012.

[4] G. Rogers, "Do grades make the grade for program assessment?" *ABET Quarterly News Source*, Oct. 2003. [Online]. Available: http://irap.appstate.edu/sites/irap.appstate.edu/files/do-grades-make-the-grade.pdf

[5] J. K. Estell, "Streamlining the assessment process with the faculty course assessment report," *International Journal of Engineering Education*, vol. 25, no. 5, p. 941, 2009.

[6] S. Verma, "Matrix based approach to assessment of an educational program along abet criteria," in *American Society for Engineering Education Annual Conference and Exposition*, Portland, OR, 2005.

[7] M. Dettman, "Effective, efficient, direct assessment of programmatic outcomes," in *American Society for Engineering Education Annual Conference and Exposition*, Portland, OR, 2005.

[8] E. T. Services. (2015, Mar.) Major field acheivement test. [Online]. Available: https://www.ets.org/mft/about/

[9] C. E. E. Board. (2015, Mar.) Sat website. [Online]. Available: https://sat.collegboard.org/home

[10] E. T. Services. (2015, Mar.) Gre website. [Online]. Available: http://www.ets.org/gre

# Assessment Methodologies for Abet Accreditation: Success Factors and Challenges

**Prof. Nabil El Kadhi[1], and Dr. Minerva M. Bunagan[2]**
[1]Deputy Vice Chancellor For Academic Affairs, UoB, Buraimi, Oman
[2]College of Engineering, UoB, Buraimi, Oman

**Abstract –** *This paper presents the importance of a well-established assessment and evaluation process in leading to an improved program delivery. It details a University experience in employing a combination of direct and indirect assessment methods to determine the Program Educational Objectives (PEO) and Students Outcomes (SO) attainment for its Bachelor of Science in Computer Program. In its 3-year cycle of PEO and SO assessment and evaluation, various success factors were identified and at the same time challenges were uncovered. The success factors, as detailed in this paper, include appropriateness and coherence of the course – SO – PEO mapping and the nature of the assessment process, careful design of the assessment tools and instruments, involvement of stakeholders in the assessment process, availability of resources, periodicity and effective documentation of the assessment process, and utilization of assessment information and dissemination of evaluation results. While these factors led to a successful program delivery, the College identified an opportunity to further improve its assessment and evaluation process by using a set of courses to assess a set of specific student outcomes that may lead to an improved collection of assessment information and generation of meaningful information.*

**Keywords:** Outcome Based Education, Learning Outcome, Direct and Indirect Assessment, Measurement and Evaluation, Accreditation.

(*) **The paper presented an experience in a University having an Abet accredited program and does not refer to UoB experience**.

## 1 Introduction

The traditional education system which is primarily focused on determining student performance and achievement based on what is taught was shifted to outcomes-based education (OBE) in early 1990s. This shift paved way to the value of assessment in colleges and universities as OBE focused more on student achievement of goals or outcomes, and the attainment of these outcomes are determined through a set of assessments. Assessment includes a variety of methods that are used to determine the readiness, progress, and skills acquired by a learner. The techniques and strategies that are utilized by any educator vary depending on the goal of assessment. Often, assessments are done to determine the weaknesses of the learner and provide the necessary intervention; and generally, assessments are done to determine the performance of the learner in a specific activity or learning process. There are 3 phases of assessment - planning, implementation, and improvement and sustenance of the process. Various stakeholders should be involved in assessment and the purposes of assessment should also be identified. Assessments can be done in various levels: program, student and course level and many are involved [1]. The assessment process is guided by an assessment plan for a successful implementation. During the entire assessment process, collection, organization and analysis of assessment information should be done to arrive at findings that need to be examined; disseminated and necessary actions should be done when needed. Assessment approaches include summative and formative assessments. Summative assessments provide judgments about the manner and value of the assessment based on a certain performance standards set, and when these are done repeatedly, the results can be utilized to improve the program. Formative assessments provide feedback that can be used to modify, form and improve the program. However, both are useful not only for overall program performance but also for student performance [2].

Various studies [2, 3, 7] reveal that learning outcomes are assessed using a combination of both direct and indirect assessments. A certain study presented easy and effective methods of assessments which are – (1) the Assessment day concept; (2) Personal class assessment; (3) sampled student work; (4) faculty interview groups; and (5) student focus groups. The study in [3] revealed that the faculty takes a greater role in assessing learning outcomes and that the methods considered both the direct performance of the students and indirect assessments, considering the perception of the faculty and the students in the attainment of learning outcomes using interviews and focused group discussions. In another study of learning outcome assessment [1], the assessment which was done in 3 levels, program level, student level and course level, involved faculty members in the development of course-level assessments. The same study specifically mentioned that course level assessments utilize a certain number of courses, whereby a set of questions were formulated per topic to assess a certain learning outcome. The types of questions asked focused on skills which represent the basic knowledge obtained by the students at the end of the course. Further, the same study also mentioned that faculty members meet and make decisions at the end of each year to identify areas to improve. The feedback loop allowed them to determine the assessment methods that also need to be improved. The study [4] presented how an assessment tool which considers peer assessment and individualized assessment is used in a capstone project. It aimed to introduce a tool that assists the instructor in evaluating student performance in capstone projects. The assessment tool was used by a sample set of students and following it, the students were asked regarding their perception on the use of the assessment tool. It turned out that generally, the students are satisfied with the assessment tool. However, the constraint is that friendship is found to have an effect on the assessment result. The tool is used to mark the performance of the students based on the group output, considering their individual effort and contribution to the group.

As mentioned in the earlier paragraphs, assessments are conducted to obtain data and be able to establish the performance of the program and the students/ learners. Any University seeking accreditation or interested to obtain program confidence rely on the attainment of student outcomes as basis for program improvement. Hence it takes a serious look in the entire assessment process. One of the criteria for Abet accreditation is a documented and effective periodic review of program educational objectives (PEOs) and student outcomes (SOs) [5]. This is achieved through a well-established implementation of assessment process, which is set and designed by the University itself. This study relates the assessment experiences of one of the Universities which obtained Abet accreditation for its Bachelor of Science in Computer Science Program. It describes the various assessments implemented in the University - program level, student level and course level. In addition, it discusses the success factors and challenges encountered in the assessment process, and provides recommendations and actions to address the challenges.

# 2 Abet Accreditation Requirements

The Abet criteria for accrediting computing programs have two sections – general and program criteria. The general criteria apply to all computing programs accredited by an Abet commission, and include students, program educational objectives, student outcomes, continuous improvement, curriculum, faculty, facilities and institutional support. On the other hand, the program criteria include additional requirements specific to the Computer Science Program under student outcomes, curriculum and faculty. Each of the mentioned criteria specifies a certain set of requirements that must be satisfied by the program in order to obtain accreditation [5].

# 3 Assessment and Abet

There are two important elements that apply to the mentioned criteria for accrediting computing programs. These are assessment and evaluation, and they are both applied to Program Educational Objectives (PEOs) and Student Outcomes (SOs).

Programs that seek or undergo accreditation by Abet have a certain set of program educational objectives (PEOs) and student outcomes (SOs), which are assessed and evaluated based on the policies and procedures of the University. The policies and procedures which are implemented university- or college- level include the 2 components for both the PEOs and SOs; namely: (1) Method of Assessment; and (2) Method of Evaluation. These are specifically discussed in the paragraphs that follow.

## 3.1 Program Educational Objectives (PEOs)

The PEOs are based on the needs of the program's constituencies, which include both the alumni and employers as the primary constituencies and the faculty and students as the secondary constituencies. The Assessment and Evaluation of the PEOs are discussed below:

### 3.1.1 PEOs Method of Assessment

Indirect Assessment is employed for the PEO, where a questionnaire is used as the instrument to obtain data.

Procedure: PEO Assessment is administered towards the end of each year for each batch of alumni after 3-5 years of graduation, based on the policy and procedures of the University. The policy and considering the local market and corporate environment as well as the best international practices in that regard, set 3 as the minimum years based on the Alumni Tracer Study Report, where the graduates are able to land on a job at an average of 1-2 years. As a matter of fact, there is no ideal duration to consider after the graduation, however factors of market understanding, corporate requirements and accommodation phases are considered. Hence we believe that the number of years after graduation, required to have an adequate measurement of the alumni efficiency that reflect a true measurement of their capabilities is obtained after a period of years that is somehow equal to a normal study duration which is in our case 3-5 years. Specific assigned office conducts the assessment through a questionnaire upon the request of the college offering the program. The office identifies the respondents and the sample size. Standard statistics requirements and best practices are to be observed here. The sample size should be big enough and is in general not less than 1/3 of the total considered batch of alumni. Common sampling techniques [6] are being implemented. In addition special care is given to assure that the sample is significant of all alumni categories as per their GPA (high, average and low)

Assessment Instruments: The PEO Assessment uses a questionnaire for the alumni and the employers. Considering that the PEOs reflect what the graduates have to attain after years of graduation and that they should be based on the needs of the constituencies, the alumni and the employers being the primary constituencies are identified as the respondents. The alumni, as the product of the program, reflect and represent the success of the program. The employers of the graduates provide input regarding the performance of the alumni and where the industry will be in the future. The questionnaire includes a set of items that allow the respondents to determine the attainment of the PEOs through a set of indicators. The indicators reflect how the alumni are able to demonstrate success in the program. The alumni- and employers-respondents provide their perception on the attainment of the PEOs using a scale of 1 to 5, with 5 as the highest.

Samples: The respondents of the PEO Assessment are the constituencies of the program – the alumni and the employers of the alumni-respondents. Ideally, the University targets to have a total enumeration for its respondents. However, since it is still building its database and locating/ tracing the alumni is a hurdle, it adopted a convenient sampling method, where all the available alumni and employers for the specific batch are considered as respondents. Common sampling techniques [6] are being implemented. In addition special care is given to assure that the sample is significant of all alumni categories as per their GPA (high, average and low)

Communication: The collated data by the concerned office are submitted to the college for further analysis and interpretation.

#### 3.1.2 PEOs Method of Evaluation

Analysis and Interpretation: A committee composed of senior faculty members selected by the college is in-charge of analyzing the data and interpreting the results. The committee members are in general chosen considering a set of criteria including years of experience in the college, previous experience in quality assurance related aspects and critical thinking capabilities. Most importantly, all committee members should be participating actively into the program delivery from various levels (year one to year four) and with various curriculum components (core courses, general education courses and elective components). Having a mixture of all those, delivery participants in the committee make sure that the analysis is as comprehensive as possible and consider various curriculum properties. The committee decides on the expected level of satisfactory attainment and the weight allocated for each of the surveys in order to determine the combined contribution of the surveys in the attainment of the PEOs. With the rating scale for the survey instruments from 1 to 5, where 5 is the highest, the committee set the expected level of satisfactory attainment to 3.0/5. Moreover, the committee also set equal percentages, 50% each for the alumni and the employer survey, considering that both are primary constituencies. Based on the combined results of the alumni (50%) and employer (50%) survey, the committee compares the result to the expected level of satisfactory attainment and concludes the attainment of the PEOs. The analysis and interpretation of the data are expressed in a form of a comprehensive report. The report includes suggestions and recommendations to be considered part of the continuous quality improvement process.

Communication: The PEO Evaluation Report is submitted by the committee to the College for Approval. The result is disseminated to the stakeholders in different ways, such as through consultative meeting with the Program External Advisory Panel, alumni general assembly, or sent through mail and published in the University Website. The College addresses any identified deficiency or weakness through the improvement plan and its implementation is also monitored by a specific committee for continuous quality improvement college level. The PEO Evaluation Report is utilized for the continuous improvement of the program. While changes in the program take place within a 3-year cycle, the findings and recommendations based on the result of the PEOs are collated and considered in the next program revision cycle (as per university rules and regulations).

### 3.2 Student Outcomes (SOs)

The attainment of the SOs is based on a set of data which are obtained through direct and indirect methods from various stakeholders. The assessment and evaluation of the SOs are discussed below.

#### 3.2.1 SOs Method of Assessment

SO Assessment is done through direct and indirect methods. As Abet does not prescribe a specific method of assessment, the University decides on the method to implement. The University set a certain percentage to each type of assessment method, 60% for direct and 40% for indirect. More weight is given to direct assessment method, as it obtains data using various course assessment instruments (test/ exam/ course projects) and specific rubrics are used to evaluate if a student outcome is achieved or not as compared to the indirect assessment method that uses survey instruments where the respondents provide answers to the specific questions included in the survey as a base of the indirect assessment.

The **direct assessment** of SOs through Courses is done through the analysis of the Course Intended Learning Outcomes (CILOs). All the courses offered for the concerned semester which are mapped to the specific SOs are considered in the assessment and evaluation. A case on the direct assessment of SOs is discussed in Section 4 and the procedure is discussed below:

The direct assessment through courses is done by the faculty member through the assessment of the CILOs. The assessment of the CILOs utilizes 4 course elements; namely: (1) Course Assessment Plan, (2) Table of Specifications (TOS), (3) Student Summative Assessment (test/ exam and course project) and (4) Course Evaluation Report.

The Course Assessment Plan is a table showing the method of assessment of each CILO, the performance rating, the rubrics used, and the contribution weight assigned to the specific CILO in the entire course. As mentioned by Banta and Palomba (2015), an assessment plan is needed to carry out successfully an assessment process [2]. Also, rubrics should be there to provide objective marking [7, 8].

The Table of Specification (TOS) is a table that shows the mapping between the questions, the learning domains based on Bloom's taxonomy, the course topics, assessed CILOs and assigned marks. This leads to a well-designed student course assessment instrument that considers both the comprehensiveness of the exam and the appropriateness of the exam to the level of the students as per the identified assessed ILOs and covered topics.

Based on the TOS, the student course assessment instrument, (test / exam) is designed. The exam sheet and/or the TOS include a clear breakdown of the marks by question to rubrics or marking criteria. The use of clear rubrics helps improve the performance of the students [7, 8]; and allows the course coordinator to fairly mark student outputs, and ease the moderation process.

The data for the preparation of the Course Evaluation Report are taken from samples of student assessed work. Based on the University guidelines, 15 is set as the minimum size of the sample for classes with more than 15 students and full cohort is considered for classes having fewer students. The declared size is an acceptable representation as it already provides a probability that there is at least 1 student in the sample who has not met the requirement [3]. The sample indicates how a CILO is attained and does not assure that all the CILOs are attained by the students. The Course Evaluation Report takes its inputs from the CILO worksheet considering the marks of students in the sample. The CILO worksheet is used to obtain the percentage of attainment of each of the CILOs by the students by entering the marks of all students included in the sample for each of the assessments as specified in the Course Assessment Plan. The weight point average of each CILO is used to obtain the SO value for which the CILO is mapped. This value, ranging from 1 to 5, with 5 as the highest is considered as input in the analysis of SO attainment by direct assessment method through the courses. The use of the direct assessment, through mean of student performance, where all scores of the student in the sample size are considered is generally used by Universities in assessing learning outcomes [7].

The **indirect assessment** of SOs obtains data using a questionnaire administered to specific respondents. It has 3 components; namely: (1) senior exit survey; (2) self-assessment of SOs by students enrolled in research projects; and (3) SO Evaluation by Practicum Supervisors. Each component is assigned a percentage of contribution to the attainment of SOs. As mentioned above, the indirect assessment method contributes 40% to the SO attainment; and this is distributed as 10% each for the senior exit survey and self-assessment of SOs by students enrolled in research projects and 20% for the SO Evaluation by Practicum Supervisors. Greater weight is assigned to the SO Evaluation by Practicum Supervisors since the perception is coming from an external body, eligible to give an assessment about the students' performance. The chosen percentage reflects the university estimation of each factor and might be adjusted from one implementation case to the other. Those percentages have been agreed on after consultation with the college members and by looking to other institution experiences. The university considered also the program and course external examiners who confirmed the validity of the chosen percentages and advised the college to evaluate the obtained results of couple of cycle of implementation in order to determine if any elements require revision. Below, we include a detailed description on the procedure for each type of assessment:

The senior exit survey is administered by the Guidance Office to all the graduating students of the program every semester. The graduating students accomplish a questionnaire, which has 4 subsections, namely: General Information; Experience in the College; Advising Services, Curriculum and Instruction; and Facilities and Academic Infrastructure. The subsection – Experience in the College contains the data on the attainment of the SOs, hence, this specific sub-section is considered only for the SO assessment. The cited sub-section contains statements on how the students' education contributed to the achievement of the student outcomes. The graduating student-respondent specifies the extent of attaining the SO using a value from 1.0 to 5.0 with 5.0 as the highest. The senior exit survey, as mentioned in the previous paragraphs contributes 10% to the over-all attainment of SO evaluation.

The self-assessment of SOs by students enrolled in research projects is administered by the faculty handling the Research Project course. The research project can only be enrolled by students who have taken the required pre-requite courses and is registered in a senior year (third or fourth year) of the program. The research project (or capstone) is considered as one of the major requirements for graduation. The course allows the students to integrate all the theories learned with practical skills by requiring them to work on designing, implementing (realizing) and documenting a project. Hence, through this course, it is expected that all the student outcomes are attained by the students. Through the questionnaire, the student enrolled in the research project course determines his extent of the attainment for the SOs using a value from 1.0 to 5.0 with 5.0 as the highest. The faculty provides the collated data to the college for further processing and analysis. Same as the Exit Survey, the self-assessment of SOs by students enrolled in research projects contributes 10% to the over-all attainment of the SOs.

The SO Evaluation by Practicum Supervisors is administered by the Practicum Instructor (university faculty taking care of the face to face sessions with the students) to the Practicum Supervisors (company side) every semester. The practicum course exposes the students to a practical work experience along their field of specialization towards the end of the completion of the Bachelor of Science in Computer Science Program. The student undergoes a practical/ on-site training in the company for 240 hours. The survey questionnaire which is accomplished by the Practicum Supervisor contains items related to the SOs. Through the questionnaire, the Practicum Supervisor indicates the extent of SO attainment for the student using a value from 1.0 to 5.0 with 5.0 as the highest. The SO Evaluation by Practicum Supervisors contributes 20% to the over-all attainment of the SOs.

#### 3.2.2 SOs Method of Evaluation

A committee composed of faculty members selected by the college is in-charge of analyzing the data and interpreting the SO attainment results. It is the committee which decides the expected level of satisfactory attainment and based on the percentage assigned to each of the method of assessment, the committee determines the combined contribution of the assessment methods in the attainment of the SOs. With the rating scale for the survey instruments of 1 to 5, where 5 is the highest, the committee set the expected level of satisfactory attainment to 3.0/5. The committee obtains the SO Assessment results and makes an analysis of the results every semester. The SO Evaluation Report is submitted to the college for approval. Based on the findings and recommendations, the college prepares an improvement plan. Any deficiency or weakness, such as adding more student assessments in the course; identified in the report that necessitates immediate action is carried out by the college. At the end of the School Year, the committee makes an annual analysis of the SO assessment results and submits a report to the college. The SO Evaluation result is disseminated to the stakeholders in different ways, such as through consultative meeting with the Program External Advisory Panel (PEAP), alumni general assembly, or sent through mail and published in the University Websites.

## 4 Success Factors and Challenges

The University's experience in obtaining Abet accreditation brought a lot of improvements in terms of program delivery. With a focus on attaining student outcomes, in the program, student and course levels, the University ensured that all methods of assessment and evaluation for both the PEOs and SOs are in-place and properly implemented. The College, in its interest to obtain Abet accreditation tailored all its academic processes and procedures to Abet requirements. It is worthy to note that several factors contributed to the successful assessment process. Among these are the following:

### 4.1 Appropriateness and Coherence of the Course – SO – PEO Mapping and the Nature of the Assessment Process

The correctness of the mapping greatly affects the attainment of the outcomes. When the course, through its CILO is not correctly mapped to the SO, the SO may not be attained. In the same manner, when the SOs are not properly aligned to the PEOs, the successful assessment of PEOs cannot be guaranteed. Within the 3-year implementation of the assessment, changes in the mapping of the Course to SOs took place during the second cycle based on the CILO analysis. It could be noted that during the first cycle of implementation of the SO Assessment and Evaluation, the program implemented the direct assessment of SOs through a selected set of computing courses. Using the established University policy and procedures for SO Assessment and Evaluation, 3 courses were identified to be used to determine the attainment of each SO. The CILOs mapped to the specific SO were assessed and the assessment methods vary from one course to another. Considering all the computing courses is the enhancement adopted during the 3rd cycle of SO assessment. The previous 2 cycles assessed the SOs considering only 3 computing courses for the direct assessment. The enhancement aimed to have a more accurate evaluation as it includes more relevant curriculum component covering various computing areas. In addition, program external examiner advised the College to consider a larger scale of courses to be included for the SO Assessment. Based on the previously mentioned element, and based on the recommendation of the Faculty Committee for SO Assessment and Evaluation, the College Council approved the inclusion of all the Computing courses in the Direct Assessment of the SOs. As a result, of the total number of SOs, 64% of the SOs have a better attainment during the 3rd cycle of evaluation, as manifested in the increase of values during the 3rd cycle. At an average, the increase in the SO result from one cycle to another is about 20%. On the other hand, there are 36% SOs which decreased its values at an average of 30% during the 3rd cycle evaluation. The comparative analysis of the 2 cycles where only 3 computing courses were included in the first cycle and all computing courses for the 3rd cycle shows globally a stability in the SO attainment (through direct assessment, with an average of 3.70 and 3.85 for cycle 1 and 2 respectively) with a clear trend of increase. This is a manifestation of adequate SO direct assessment tools as well as an indicator that the implemented changes whether in the exam scheme, course material or teaching methodologies are appropriate. The increase of 64% in SOs dealing with the students' abilities to apply and use current techniques efficiently in answering professional requirements, with the recognition of the need to engage in a life-long learning is an achievement that the college is proud to note and will consider as a motivation for further improvement.

### 4.2 Careful design of the assessment tools and instruments

All the documents and reports used from the collection of assessment information to the dissemination of the evaluation results should be appropriately designed. All instruments that were used in direct and indirect assessment method went through a 3-step design-verification process. This ensured the validity of the assessment instruments. One of the important features of the assessment instrument is the presence of clear rubrics, which as discussed in the earlier section, served as a tool to improve performance.

### 4.3 Involvement of Stakeholders in the Assessment Process

The University recalls its experience from the formulation of the PEOs, SOs and CILOs to the utilization of evaluation results, where many stakeholders are involved. The responsibilities of stakeholders vary from one phase to another considering the 3 phases of assessment – planning, implementation, and improvement and sustenance of the process [2]. In the planning phase, the faculty members are directly involved as they designed the assessment plan for the course-level assessment; and a committee composed of faculty members are also involved in collecting PEO and SO assessment information and analyzing the results. During the implementation phase, faculty, students, alumni, employers, and practicum supervisors are involved, either as respondents, as source of direct data or in-charge of gathering the assessment information. The same stakeholders are involved in the improvement and sustenance phase, along with the Program External Advisory Panel (PEAP) who serves as the representatives of the primary program constituencies. The PEAP consists of five (6) members, three (3) employers, one (1) representative of the BSCS alumni and two (2) representative from professional organization. The PEAP members have been selected based on their qualifications and prospective contributions to the growth and development of the program.

### 4.4 Availability of Resources

One of the most important aspects of assessment is the availability of assessment experts and the presence of an automated learning outcome calculation. The use of an automated system ensured the accurate calculation of assessment data, and at the same time reduced the efforts of the faculty [9]. The college developed a semi-automated tool based on the usage of embedded excel computation formula in order to both assure coherence and consistency of the obtained results and also to facilitate and support teachers and various participants to the evaluation process.

### 4.5 Periodicity and Effective Documentation of the Assessment Process

The assessment process is one of the established academic activities in the University. The assessment and evaluation of the PEOs are done every year. The assessment and evaluation of the SOs and CILOs are done every semester and the results are summarized annually. In every process, a committee is involved and tasked for a responsibility starting from the collection of data to the analysis and dissemination of results as well as in addressing findings.

### 4.6 Utilization of Assessment Information and Dissemination of Evaluation Results

The primary purpose of assessment is to evaluate student performance and improve program delivery. As a result of the first cycle of evaluation, significant changes in the program, which include, adding Mathematics and Computer Courses in the curriculum to improve the problem solving skills of the students, updating textbooks, revising lecture notes and improving assessments took place. The results of the second cycle of evaluation were used to enhance the Bachelor of Science in Computer Science courses and served as basis in the integration of embedded assessments in most of the computing courses. The key indicators for assessing these courses have been improved to match the outcomes being assessed and make the process more consistent and effective. In fact, the following recommendations were made based on the result of the SO evaluation for the second cycle:

- Review the ILOs of the courses, specifically those which are mapped to SOs a, g and k, as some may not have addressed properly the specific SO to which they are mapped;
- Review the teaching and assessment methods of the courses. Use rubrics and include in the exams more questions that develop critical thinking. Strengthen problem-based learning by having a project as major component of advanced courses; and
- Include more than 3 courses in the evaluation of SO by direct assessment.

The various factors that contributed to the success of the assessment process led to the improvement of program delivery as mentioned above. However, the early stages of implementation brought also some challenges as well, starting from the awareness of the faculty about student outcomes, formulation of student outcomes and methods of assessments and evaluation, to the design of the assessment instruments, utilization of results and to the sustenance of the processes and procedures. One among the major challenges when dealing with outcome assessment is to find the adequate equilibrium between the number of assessment and the assessed ILOs as well as the adequate periodicity, time frame and level. In fact, some practices focus on a relatively high number of assessments (mainly in first and second year) as a way to ensure assessing targeted basic concepts in separate way. Such - even defendable - will lead to a lack of a global overview on the learner capability to integrate and combine knowledge and will present an additional challenge of sustainability considering the required efforts and time by the faculty to achieve such. Moreover designing a well equilibrated assessment that allows to track students level and allow to identify various level of achievement is a major concern and we have seen often time a need for specific capacity building session in that regard. The assessment as presented here is now far from being an isolated action for measuring a single or unique performance. It has to be seen in a more complete vision of ILOs achievement. Such 'snapshot' of efficiency measured by one assessment has to be included in a full evaluation process. Hence nowadays, educators are having more challenging task in doing such and their – relationship- with one evaluation or assessment will be for a longer period than just designing and conducting the assessment. Educational, Contextual, Organizational and Time constraints factors have so to be considered when designing assessments, deciding on assessment components tools and methodology in order to ensure correctness, fairness, sustainability and adaptability in a dynamic environment. The dilemma of choosing adequately the number and category of assessments will be usually of a major interest as increasing the number and frequency of assessment does not indicate nor assure that student outcomes are attained. In fact, Abet does not recommend that the program assesses student outcome every year to know how well it is able to attain student outcomes. Rather, it suggests cycles of assessments using performance indicators using 2-3 core upper-level courses mapped to the outcomes, on a rotation basis [10].

# 5 Conclusions and Recommendations

The University's established assessment methodologies led to the collection of meaningful data which were utilized to improve program delivery. While the assessment methods employed improved the Bachelor of Science in Computer Science program, this does not mean that the existing assessment methods employed are efficient enough, considering that there are excessive assessments done every period and that assessment is part of the responsibilities of the faculty. Hence, the college should explore the use of a set of courses for a specific set of SOs by semester, within a 3-year cycle. This should be done to identify a better assessment methodology with the objective to simplify the assessment processes without compromising the quality of results. The experience of rethinking the assessments and the assessment process within the global Abet framework and requirements definitely has a positive impact on the program delivery improvement. The set of choices, components and process of ILOs assessment as presented in this paper have to be considered as – contextual – and subject to continual evaluation and improvement. As a matter of fact we do believe that this is a very intermediate step before reaching a more equilibrate, sustainable and efficient assessment methods and processes.

# 6 References


[1] The Assesment of Learning Outcomes in Information Assurance Curriculum. N. Paul Schembari, Mike Jochen. Information Security Curriculum Development Conference 2013 October 12, 2013, Kennesaw, GA, USA. Copyright 2013 ACM 978-1-4503-2547-9-0/10/1

[2] Planning, Implementing, and Improving Assessment in Higher Education. Second Edition. Trudy W. Banta, Catherine A. Palomba. Copyright © 2015 by John Wiley & Sons, Inc

[3] Five Easy but Effective Assessment Methods. Dick K. Blandford and Deborah J. Hwang. SIGCSE'03, February 19-23, 2003, Reno, Nevada, USA. Copyright 2003 ACM 1-58113-648-X/03/0002

[4] Fair Assessment in Software Engineering Capstone Projects. Adelina Basholli, Fesal Baxhaku, Dimitris Dranidis, Thanos Hatziapostolou. BCI '13, September 19–21, 2013, Thessaloniki, Greece. Copyright 2013 ACM 978-1-4503-1851-8/13/09

[5] Criteria for Accrediting Computing Programs, 2015-2016. http://www.abet.org/accreditation/accreditation-criteria/criteria-for-accrediting- computing-programs-2015-2016/

[6] Population and Sample. Sampling techniques. Paula Lagares Barreiro and Justo Puerto Albandoz. Management Mathematics for European Schools. 94342 - CP - 1 - 2001 - 1 - DE - COMENIUS - C21

[7] Meaningful Assessment. Geoff Stoker, Jean Blair, and Edward Sobiesk. SIGITE'14, October 15–18, 2014, Atlanta, Georgia, USA. Copyright © 2014 ACM 978-1-4503-2686-5/14/10. http://dx.doi.org/10.1145/2656450.2656478

[8] Application of rubric in learning assessment: A proposal of application for engineering students. Ana M. Vivar Quintana, Ana B. González Rogado, Ana B. Ramos Gavilán, Isabel Revilla Martín, María_ Ascensión Rodríguez Esteban, Teófilo Ausín Zorrila, Juan Fco. Martín_ Izard. First International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'13), November 14 - 15 2013, Salamanca, Spain Copyright 2013 ACM 978-1-4503-2345-1/13/11…http://dx.doi.org/10.1145/2536536.2536604

[9] Selective and Objective Assessment Calculation and Automation. Chunlei Liu and Li-Mei Chen. ACMSE'12, March 29–31, 2012, Tuscaloosa, AL, USA. Copyright 2012 ACM 978-1-4503-1203-5/12/03

[10] Module 4 – Quality Improvement of Student Learning. http://www.abet.org/network-of-experts/for-current-abet-experts/program-evaluator-candidate-training/module-4-quality-improvement-of-student-learning/

# SESSION

# CASE STUDIES, GENERAL AND CAPSTONE PROJECTS TO ENHANCE LEARNING + BRIEF SURVEYS

## Chair(s)

**TBA**

# Project-based Learning in Introductory Biostatistics Using Excel

*Xiaohui Zhong*

Mathematics, Computer Science, and Software Engineering, *University of Detroit Mercy, USA*

***Abstract*— Abstract— Students in biostatistics classes often tend to believe statistics/biostatistics is both difficult and irrelevant to their study. Previous studies suggest project-based learning in statistics courses may enhance students' attitudes toward statistics. Biostatistics courses taught using project-based learning approach can give bio-science students firsthand experience in data design and collection, mastery of statistical reasoning skills and the ability to select appropriate procedures for analyzing data which will prepare them for further studies and research. This paper describes a first semester undergraduate-level, project-based biostatistics course designed to cultivate these skills for bio-science students at the University of Detroit Mercy. Discussion focuses on the rationale for adopting and the design of the projects as well as the outcomes of the projects. Student evaluations indicated that project-based learning improves understanding of biostatistics overall.**

***Index Terms*— Excel; Biostatistics; Project-based learning; Statistical reasoning; Teamwork;**

## I. Introduction

Biostatistics is not simply a non-organic combination of biology and statistics. It is statistics applied in the context of biology, health science, life science, etc. [9]. As more advances emerged in the fields of life science and bio-science, the knowledge and ability to apply bio-statistical techniques in research and practice has become increasingly important. The professionals and researchers in bio-science in this new age will need greater ability to evaluate, interpret, and apply the information and data from practice and experiment than any time before. A good understanding of biostatistics can improve critical thinking and decision making in data analysis and research. For all these reasons, biostatistics is now considered as an essential tool in planning, conducting experiments, and delivering results in bio-science. On the other hand, because more complicated statistical methods are being reported in the medical, biology, and other scientific literature, lack of training in biostatistics will likely increase difficulties for many researchers [10], and [14]. Thus, effective bio-science programs usually include biostatistics training in their curricula to successfully prepare students for this important lifelong learning skill. Especially, almost all the medical institutes and universities worldwide have provisions to teach biostatistics to undergraduate and graduate students [11].

X. Zhong is with the Department of Mathematics, Computer Science, and Software Engineering, University of Detroit Mercy, Detroit, MI, USA (phone: 313-993-1013; fax: 313-993-1187; e-mail: zhongk@ udmercy.edu).

At the University of Detroit Mercy (UDM), biostatistics is a required second year course for all students in pre-medical, pre-dental, physician assistant, nursing, biology, and related majors. This is a one-semester course covering brief sampling theory, data presentation, topics of descriptive statistics, odd ratios, Bayes' theorem and diagnostic test/ROC curves, mortality tables, and inferential statistics such as confidence interval in different contexts, a variety of parametric/nonparametric hypothesis testing, simple linear regressions, correlation, and one way ANOVA. The course content agrees the top ten topics in biostatistics offered by US medical schools [10]. The book, Principle of Biostatistics by Pagano & Gauvreau [9] was adopted as the main text book supplemented with instructor notes. Students are required to use Excel to analyze their data.

## II. WHY PROJECT-BASED LEARNING?

Students frequently view statistics as the worst course in college [6], let alone biostatistics. UDM students in the pre-medical, pre-dental or bio-science programs take this course to fulfill curriculum requirements. They usually lack motivation and do not see it as relevant to the work of their major and other courses [10]. In order to break down the walls of boredom and apathy, innovative approaches must be taken to engage students so they do not just accept information passively, but pursue knowledge actively. Keeping in mind that instructors should "encourage students to view the statistical process as a whole, relate the data to the context, …, and think beyond the textbook" [4], teaching biostatistics with projects became an obvious approach, because project-based learning is a comprehensive approach to classroom teaching and learning that is designed to engage and motivate students in taking active part in their learning with real world problems.

Project-based learning is not a new subject as described by number of papers tracing back to early 90's [5], and [13]. The importance and effectiveness of project-based learning in statistics were well asserted by both authors. Bennie [1] affirmed such result by the statement: "The use of project is very helpful in assisting the learning of students. Their active involvement in the task forced them to think and enhance their learning. The use of real data of their own choice motivated them because they want to know what conclusion they might come up to. The use of technology was implicit because all the data analysis was done by Minitab and the report was written in Word. Without the projects their understanding of the process of the problem solving using the statistical thinking strategy outlined would be very theoretical."

While all these papers address the importance of project-based-learning, reporting the positive effects, and describing the details of projects, all projects vary in styles. Short and Pigeon [12] emphasized a lot in the planning stage. Smith [13] spent many hours on written and oral presentations with a sequence of little projects involving collecting simple data. Nascimento and Matins [7] found open ended projects more suitable for students in graduate study, some incorporated in service-learning [8]. They all suggested great ideas in lifting up students' enthusiasm and changing their perception toward the study of statistics. These good ideas should be adapted to teaching biostatistics to meet the needs of the unique profile of the students in health and life sciences and to help students to overcome the challenges in their studies. To better serve the students in the University of Detroit Mercy, I developed projects that go along with the flow of the content of the class of biostatistics so that concepts learned in class are integrated as the projects progress.

In this article, I discuss the detail and rationale in the design of course-long projects; depict criteria for evaluating the projects; present survey results indicating that projects have the potential to help students learn; point out factors in project design that affect motivation and interest of students; examine difficulties that students and teachers may encounter with projects; and describe what improvement should be made to support students as they work on projects, so that motivation and thought are sustained.

## III. THE PROJECTS

The goal of project-based learning is to allow students to gain real world experience, enhance analytical and problem solving skills, promote team work, as well as improve writing, communication and presentation abilities. While writing instruction and examples of projects, thoughts were given to balance data collection, statistical reasoning and overall presentation. To simulate real world teamwork situation, students were randomly assigned to form teams of five to six persons so they develop skills in team working with others who are not necessarily considered "friends". Once the team is formed, they will be working together through the entire semester. Unless contested by the majority of team members, all individuals will receive the same grade for the projects.

Adapted and modified from projects described by Carnell [3], the semester-long project is divided into three parts. The first part is the proposal by students. Each team will write a proposal on what kind of project they will be doing: a survey or an experiment; what research question they will attempt to answer; and what kind of data they will collect. Detailed instructions on how to write the proposal were given accompanied by two examples. For second year students, writing a proposal for a project is still relatively new for them. Such preparation will give students ideas of what a survey or an experiment would look like and guide them through the setting up of the project since this is a critical step for a successful project [12]. In order to give students enough time to design their own project, ask their own research questions and improve the data protocol, students are given three chances to submit their proposals. The first two submissions were usually returned with feedback from me. Only the last submissions were counted toward their grade. Students submitted their proposals using Blackboard and my comments/suggestions on their work were handwritten using a stylus pen in a tablet PC. In addition to receiving written feedback via Blackboard, students also come to see me to seek further instruction. Through the interaction of students and instructor and the team work of students, the proposals evolved from very basic ideas to proposals that posed some research questions and design of multi-dimensional data collection. The minimum requirement of the data format includes at least five variables of which at least three are at the ratio/interval level. The experimental projects need at least fifty data points and survey projects need at least one hundred data points for each variable. Because of the time constraint, I asked students not to include human subjects to avoid the IRB application and approval process. The first part of the project took up the first five weeks of the semester. During this period, we covered sampling methods, level of measurements and treatment of different types of data, presentation of data, and descriptive statistics.

Once students' proposals were approved, students could start collecting their data as the beginning of part II of the project. Some of the teams may have to modify their data collection plan in the proposal due to some unforeseen reasons. For example, when one of the teams was collecting information on the serving size of the cereals, they found that the serving sizes of cereals are different from brand to brand, not only in units, but in measurements too. Since some serving sizes were in grams, some were in ounces, and some serving sizes were in cups, they had to decide whether to normalize the information or abandon the data. After the data collection, the main task in this part of the project is the data organization and presentations, as well as the calculation of all descriptive statistics. Students were using Excel to perform all the calculations and to draw graphs. Some teams used the formulas taught in class, while others used built-in formulas. I encourage them to do some calculations with formulas and check the results with built-in formulas. Students were very creative in searching the Internet to find instructions to create all sorts of graphs using Excel including various ways to draw box-plots. Students were given two chances for submission of this part of the project. The feedback to them was usually the correction of mistakes in descriptive statistics and creating more visual presentation of data. This part of the project took up about three weeks. During this period, basic probability theory, probability distributions, sampling distributions, and Bayes Theorem, diagnostic test, ROC curve, odd ratios, and life table were taught.

Finally, the remainder of the semester would be on statistical inferences. The third part of the project required them to use at least one inferential statistics technique to analyze their data and answer their research questions. In the beginning, students had no idea what to do to their data. It was interesting to hear their "oh" and "ah" while teaching these concepts and ideas that could be related to their projects. For example, after the two sample t-tests were taught, some students would ask how to compare the means of more than two populations, at which time I usually introduce ANOVA. Some were eager to learn the concepts of correlation because they were seeking correlations of some variables in their projects. Some were interested in dealing with categorical data, while others were interested in non-parametric testing. It was very encouraging to have student reactions like this.

Part III of the project also required students to write the report and conclusions according to their analysis. It was emphasized that they must write in the context of the problem. In this part of the work, the most frequent question asked by students was: what do YOU want us to write? As the instructor, I encouraged students to write enough to communicate the findings to the non-technical communities, but not too much detail to bore the readers. Students still have two chances to finalize their reports. At the first implementation of project-based learning in this course, students converted their report to PowerPoint slides and presented their projects in front of the class. At the second implementation, each team created a poster and a poster session was held. The poster session was well received by the faculty and students of the university. The attendance was over one hundred during a one hour session. Some instructors gave incentives to students to attend the session and create written reports.

The project-learning approach was implemented twice during the winter semesters of 2012 and 2013. In the two classes I taught, a total of thirteen projects were created. Two of them were empirical projects and others were surveys. All projects had basic descriptive statistics. Data presentation formats included histogram, bar graphs, pie charts, box-plots, frequency tables, etc. Analysis techniques used were hypothesis tests, confidence interval, ANOVA, multi-comparison, correlation, simple linear regression, non-parametric tests, etc. Tables 1 and 2 are summaries of all projects and techniques used in each project for both semesters.

TABLE 1
Projects of Winter, 2012

| Team Name (members) | Title | Types of project | Data Presentation Format | Inferential Statistics |
|---|---|---|---|---|
| Stain Fighters (5) | Stain Fighters | Experiment | Histograms | Two sample t-tests |
| Team Tater (6) | Correlation Between Sodium Content and Price in a Variety of Chips | Survey | Frequency Table Histogram | Confidence intervals; Correlation with hypothesis testing |
| Bookworm united (5) | Textbook price survey | Survey | Histogram Box-plots | Confidence intervals; ANOVA Multiple comparison with Bonferroni correction |
| Doctors (5) | Vehicles and How Their Appearance, Year, and Performance May Affect Their Price | Survey | Histogram Frequency tables Boxplot Bar graphs | ANOVA |
| IceCold (5) | Finding Healthier Ice Cream: Statistical Analysis of Price, Calories, Fat and Sugar Content | Survey | Histogram Frequency tables | Confidence intervals; ANOVA; Multiple comparison with Bonferroni correction |
| Meow (5) | The Price and Protein Correlation in Various Cat Foods | Survey | Descriptive statitisics | Two-sample t-tests; Confidence intervals; Correlation with hypothesis Test |
| Team-bio (6) | Chocolate Candies | Survey | Scatterplots | Confidence intervals; Correlation with hypothesis test |

TABLE 2
Projects of Winter, 2013

| Team Name (members) | Title | Types of project | Data Presentation Format | Inferential Statistics |
|---|---|---|---|---|
| Five Fabulous Females (5) | Evaluation of the Best Fast Food Restaurant | Survey | Bar graphs Box-plots | ANOVA; Multiple comparison with Bonferroni correction |
| Team University (4) | Analysis of Tuition Cost for Universities | Survey | Histograms | Two sample t-tests |

| | | | | |
|---|---|---|---|---|
| | in the United States | | | |
| Spring-break Investigators (5) | Guide to the Cheapest Spring Break Getaways in the United States | Survey | Histograms | Two sample t-tests, ANOVA |
| Drosophil Melanogaster (6) | Effects of Cancer Blocking Gene on Drosophila Melanogaster Wing Cancer | Experiment | Histogram Line graphs | Two sample t-tests |
| The Munchies (6) | Analysis of Organic Vs. Inorganic Cereal | Survey | Bar graphs | ANOVA; Non-parametric tests |
| Team Diesel (6) | Price and Consumption Analysis of U.S Petroleum | Survey | Box plots, Line graphs | ANOVA; Correlation; Confidence intervals |

## IV. ASSESSMENT AND DISCUSSION OF THE PROJECTS

Incorporating projects to any class required not only planning before classes, but also involved a sequence of work though the entire semester for both the instructor and students. In this section, we will discuss how the student projects were evaluated, problems and difficulties in doing the projects, the extra workload for instructor, and students' response/attitudes toward the project-based learning.

### A. *Assessment of the Projects*

Students accumulated points from doing the projects. Each part of the project was worth 50 points total to 150 points, which was 30% of the course grade. The grade of the last 50 points consisted of two parts: 45 points for the report and 5 points earned by showing up in the presentation/poster session to reinforce the attendance. To assess the project of each team, following criteria were used:

a. Basic information such as title, team name, individual team member's names, submission information, and file name in a required format to assess participation and attention to detail.

b. Presentation: Clear statement of goals of the project and source of data; correct units, label of axes in the graphs, title of table and graphs.

c. Methodology: appropriate statistical methods related to the content, correct application of the procedures, correction calculations, and presentation.

d. Participation: team effort and conflict resolution,

e. Proper use of software: Excel for data analysis and data presentation, Word for report, PowerPoint for presentation or poster.

However, there was no definite point system on how the points are distributed precisely to avoid students drilling into details of obtaining points instead of the overall quality of the project. In addition, the point distribution was purposely left vague to allow students to get the sense that these are the general rules for real world report/research writing. The goal is provide a platform for students to conduct experiments, collect data and practice writing so that students can develop life-long learning, thinking, and writing skills.

### B. *Discussion of Students Work*

By doing the project, students had the chance to realize that data are not necessarily cookie-cut like the textbook examples. Because they had to identify appropriate procedures or models to work on their own data, students realized that data could be inter-related and multi-dimensional with various levels of measurements. Organizing data and presentation of data is a critical step of data analysis. It was emphasized that their conclusion cannot be drawn merely from the graphs; instead, any conclusion must be supported by evidence of data and analytical results.

Another concern the students had was about how much detail needs to be included in the report. At first, their concerns were how many points would be deducted if calculations were not shown. It took a while to explain to the students that they need to include enough detail and explanation to justify the point or conclusion they make, but not too detailed to bore the readers. Students are very used to looking for rubrics reflecting by their open comments. The instructor should balance how much of a detailed rubric should be included so as not to take away the creativity in writing/summary while still guiding the students through the process. It is a learning process for students to write, and also for the instructor to perfect.

Some details worth noting are units, sources of information, as well as distinction between definition and perceptions. Unit is the most frequently overlooked item. Units were either undefined, or improperly defined, or mis-aligned. When it came to terms requiring definition, students usually took casual meanings, such as "healthy", without any quantified and quality statements. They will usually consider an item with less sugar or fat as "healthy" without knowing the actual size of the item and why they are healthy. Furthermore, there was no citation, references, or sources to support such casual "saying". Clear instructions should be written to remind students about these details. Also, instructor should also constantly emphasize this during lectures.

Teamwork appeared to receive a mixed reception by students. Most teams did not have big problems with teamwork and some do like this approach. Some complaints included difficulty in scheduling meetings, non-participation, and uneven contribution. Students usually resolved the conflicts within the team except one team. For one particular team, the students had a much bigger dispute on the participation and contribution of individual members. An intervention from instructor was given to have each student write up what they had contributed to the project, proportion of participation and percentage of grade they deserved. According to their writing and input, different points were assigned to different students instead of a flat grade. More procedures have to be developed to prevent this from happening again and it is also hoped to catch the signs for early intervention.

### C. Workload for Instructor

In general, the workload for the instructor in project-based learning is heavier than regular sessions, especially without a teaching assistant. Class size of the first time was 37, which was 2 over the class limit. Class size of the second time was 34, which 4 over the class limit (under my request, class size was reduced to 30 from 35.) Each time, there were 6 to 7 project teams with 5 to 6 students per team. Instructions were posted in Blackboard and submissions were collected in Blackboard. Each semester, students had 3 submission chances for the first part of the project which allowed them to receive feedback from me and improve the project. Subsequent parts of the project had two chances for submission, and a final submission of posters or PowerPoint presentations which total to 48 grading for 6 teams in addition to the face to face consultations. Students were also required to take 3 tests and a final exam to achieve the total success of the class. Despite the workload, the achievements and compliments of the students is well worth of the effort.

### D. Evaluation of Student Attitudes toward the Projects

To get feedback from students about the project, an end of term survey was created using Blackboard survey creator. Students were encouraged to take the survey by earning 5 points extra credit. The instructor could only tell who completed the survey without knowing what each student's exact responses were, so no biases to individual students were introduced. Aggregate result of surveys of both terms is summarized in the following table.

TABLE 3
Survey Result about Attitude toward the Project (percentage)*

| | Strongly agree | Agree | Neutral | Disagree | Strongly disagree |
|---|---|---|---|---|---|
| The project connects unrelated concepts from two or more fields. | 22.9 | 42.9 | 20.0 | 11.4 | 2.9 |
| | 23.8 | 47.6 | 19.1 | 9.5 | 0.0 |
| The project helps develop excellent reasoning skills. | 14.3 | 48.6 | 22.9 | 8.6 | 5.7 |
| | 4.8 | 33.3 | 38.1 | 23.8 | 0.0 |
| The project helps dealing with real world examples outside of the classroom. | 25.7 | 57.1 | 8.6 | 5.7 | 2.9 |
| | 42.9 | 47.6 | 0.0 | 9.5 | 0.0 |
| The project provides opportunity to learn by doing. | 17.1 | 65.7 | 11.4 | 2.9 | 2.9 |
| | 14.3 | 42.9 | 19.1 | 23.8 | 0.0 |
| The project motivates my attention and increases my interest. | 8.6 | 28.6 | 28.6 | 28.6 | 5.7 |
| | 9.5 | 23.8 | 33.3 | 14.3 | 19.1 |
| The project encourages creative and critical thinking. | 17.1 | 60.0 | 8.6 | 11.4 | 2.9 |
| | 14.3 | 57.1 | 19.1 | 9.5 | 0.0 |
| The project experience improves writing/oral presentation skills. | 17.1 | 28.6 | 31.4 | 17.1 | 5.7 |
| | 9.5 | 28.6 | 38.1 | 19.1 | 4.8 |
| The project experience improves teamwork discussion and conclusion. | 25.7 | 54.3 | 8.6 | 5.7 | 5.7 |
| | 38.1 | 52.4 | 0.0 | 0.0 | 4.7 |
| Doing the project improves understanding of Biostatistics overall. | 11.4 | 48.6 | 22.9 | 14.3 | 2.9 |
| | 19.1 | 47.6 | 14.2 | 9.5 | 9.5 |

* The first row in each question represents the results of Winter, 2012, and the second row represents the results of Winter, 2013.

Results from the surveys indicated that students had positive feedback overall. Majority of students strongly agree or agree on almost all questions. The project "helps dealing with real world examples outside of the classroom", "encourages creative and critical thinking", and "improves teamwork discussion and conclusion" received the highest marked with over 70% positive opinions. Such result reveals that students really appreciated the opportunity to learn and think outside of the classroom. Given that teamwork is an emerging trend in health science and career [15], it is not hard to understand why students gave such high appraisal about this opportunity. However, project "motivate my attention and increasing my interest", and "improves writing/oral presentation skills" received relatively lower scores with only in the 30% range of agreement, demonstrating that there is still room for improvement. Finally, over 60% of students agreed that "doing the project improves understanding of Biostatistics overall." In addition, students provided many encouraging open comments and suggestions. "Working on the project was fun and quite interesting." "The project definitely does help you connect the things we learn in class to the real world," "The project parts were broken up at the right times based on when we learned the information needed to complete that part," are few of the comments posted by the students.

## V. Final Thoughts and Future Development

Overall, by having students collecting data in real-life situations, analyzing data and drawing conclusions through Excel, project-based learning in biostatistics helps bio-sciences students develop critical thinking skills, deal with teamwork problems, gives them opportunities to handle situations outside of the classroom and apply alternative methods of analysis, with the ultimate goal of motivating students to take responsibility in learning. The approach used also stimulated students to understand more about statistics and changed the attitude that everything important about statistics should be learned passively in classrooms.

The titles of students' projects showed that students did not do much investigation on the subject of biology or health science. A couple reasons can be associated with such a result. First, since this is only a sophomore class, many students may not have enough experience in the experiments related to their majors yet. Second, obtaining biological or health related data requires substantial efforts including privacy issues, IRB approvals, and lab equipment. For the future, it is recommended to change this course to junior level class so that students would gain a lot more experience in the experimental classes in their major before taking this course. Another recommendation is to have faculty with statistics/computational background to co-teach with a faculty with bio-science background to complement each other. In this way, the method and experience gained from doing the projects will benefit the students in future course work and research. Furthermore, data can be collected in a virtual fashion from a virtual environment "Island" [2] to avoid the privacy issues or IRB approval procedure.

Finally, various computational tools can be more practical and more relevant. To accomplish this, useful computational statistics software, specifically tailored to bio-science data analysis projects, will be used. These software tools, such as BMA-CRM Simulator, CONFINT, and Dose Schedule Finder, are extremely useful for bio-science and are available for download from the website provided by the Department of Biostatistics of the University of Texas (https://biostatistics.mdanderson.org/SoftwareDownload/).

Student evaluations showed satisfaction with the approach of project-based learning. This course will continuously be revised, improved, and enriched to increase students' attention, interest and involvement. It is hoped to promote students self-directed and life-long learning. (Sample student posters are available upon request).

## REFERENCES

[1] N. Bennie, Using projects to encourage statistical thinking, Proceedings from ICOTS 6: *The Sixth International Conference on Teaching Statistics*, South Africa, 2002

[2] M. Bulmer, and K. Haladyn, Life on an Island: a Simulated Population to Support Student Projects, *Technology Innovations in Statistics Education,* 5(1), 2011, Retrieved from http://www.escholarship.org/uc/item/2q0740hv.

[3] L. J. Carnell, The effect of a student-designed data collection project on attitudes toward statistics, *Journal of Statistics Education*, 2008, Retrieved from http://www.amstat.org/publications/jse/v16n1/carnell.pdf.

[4] B. Chance, Components of Statistical Thinking and Implications for Instruction and Assessment, *Journal of Statistics Education*, 2002, Retrieved from http:// www.amstat.org/publications/jse/v10n3/chance.html.

[5] S. Fillebrown, S. Using projects in an elementary statistics course for non-science majors, *Journal of Statistics Education*,1994, Retrieved from http://www.amstat.org/publications/jse/v2n2/fillebrown.html.

[6] V. R. Hogg,, Statistical Education: Improvements are badly needed, *The American Statistician*, 45, 1991,pp. 342-343.

[7] M. Nascimento andA. Matins, Teaching and learning of statistics: the project approach, *11th International Congress on Mathematics Education*, Mexico, 2008, Retrieved from http://tsg.icme11.org/document/get/483.

[8] E. D. Nordmoe, Service-learning in introductory statistics at Kalamazoo College, *Journal of Statistics Education*, 2007, Retrieved from http://www.amstat.org/publications/jse/v15n2/nordmoe.html.

[9] M. Pagano and K. Gauvreau, *Principles of Biostatistics* (2nd ed.), Belmont, Ca: Brooks/Cole, 2000.

[10] H. Sahai, and M. M. Ojeda, Problems and challenges of teaching biostatistics to medical students and professionals, *Medical Teacher*, 21(3), 1999, pp. 286-288.

[11] W. A. Q. A. S. Sami, Biostatistics education for undergraduate medical students, *Biomedica*, 26, 2010, pp. 80-84..

[12] T. H. Short and J. G. Pigeon, Protocols and pilot studies: taking data collection projects seriously, *Journal of Statistics Education*, 1998, Retrieved from http://www.amstat.org/publications/jse/v6n1/short.html.

[13] G. Smith, Learning statistics by doing statistics, *Journal of Statistics Education*, 1998, Retrieved from http://www.amstat.org/publications/jse/v6n3/smith.html.

[14] D. M. Windish, S. J. Huot., and M. L. Green, Medicine residents' understanding of the biostatistics and results in the medical literature, *Journal of the American Medical Association*, 298(9),2007, pp. 1010-1022.

[15] A. Xyrichis and E. Ream, Teamwork: a concept analysis, *J Advanced Nursing*, 2008, Jan;61(2), pp. 232-41

**Xiaohui Zhong** received the B.S. degree from the South China University of Technology in 1982, the M.S. and Ph.D. degrees from Wayne State University, Detroit Michigan, in 1987 and 1991 respectively. She is currently a Full Professor of Mathematics at the University of Detroit Mercy (UDM), Detroit, Michigan. During her tenure in UDM, she taught courses in Probability and Statistics for engineering, Mathematical Modeling, Mathematics of Finance, and Mathematical Thinking, with publications in Cost Analysis of Light Rail Transits, Analysis of Public Transit Opinion, Collaborative Recommendation Systems, and Behavior of Distractive Driving in Dilemma Zones. She is also the Principle Investigator and Co-Principle Investigator of two major National Science Foundation grants and a grant from the Department of Transportation.

# Lattices and Boolean Algebra from Boole to Huntington to Shannon and Lattice Theory

**Hassan Farhat**
**Computer Science**
**University of Nebraska at Omaha, USA**

**Abstract -** *The study of computer design and architecture includes many topics on formal languages and discrete structures. Among these are state minimizations, Boolean algebra, and switching algebra. In minimization, three approaches are normally used that are based on equivalence relations. Partial order relations are used today in constructions of Boolean algebra.*

*In this paper we survey this important algebra from its beginnings as alternative symbolic algebra starting with George Boole and De Morgan, to Peirce, to Venn, to Huntington and to Shannon. We then look at definitions based on lattice theory. Lattices are special cases of partially ordered sets that have common properties with equivalence relations. The paper is educational in nature intended to aid the instructor on this topic from its beginnings and provide a condensed survey*.

**Keywords**: Digital Design, Boolean Algebra, Switching Algebra, Symbolic Algebra, Lattices, State Minimization

## 1 Introduction

Fundamental to all aspects of computer design is the mathematics of Boolean algebra and formal languages used in the study of finite state machines. Topics on formal languages are found in [1-3].

Boolean algebra provides the mathematical tools needed in design, realization and verification. While the algebra was intended as a mathematical tool for formalizing logic [4], many years later it found applications in computer engineering where it was adapted by Shannon as two-valued algebra to be used in design of relays [5,6]. The algebra has matured since its introduction by its founder George Boole and other scholars such as De Morgan [7]. Boole introduction of the symbolic algebra emphasized similarities and differences to traditional numerical algebra. In expanding the possible similarities between both, Boole did consider a two-valued Boolean algebra as a solution to the idempotent property ($x \cdot x = x$). The idempotent property holds true in traditional numerical algebra only for two cases, $x = 0$ and $x = 1$ [8]. Today, with few exceptions, many textbooks on computer and digital design assume a two-value Boolean algebra where the elements can be true or false, high or low, and 0 or 1 for example [9-15].

In this paper we survey the progress of this important algebra from its beginnings to today's definitions based on lattice theory. Since its introduction in 1847 the algebra has witnessed contributions by many scholar such as Pierce [16], Venn [17], Huntington [18], and Shannon [5]. We compare these contributions as we progress in the survey. After the initial idea of forming this survey, we found an expanded reference on the topic [19]. Part of our work is influenced by [19].

The paper is organized as follows. In section 2, we look at the history of the algebra from Boole to Peirce to Venn and to Huntington. Section 3 looks at the contributions by Shannon. In section 4 we look at Boolean algebra as a special type of a lattice. Section 5 discusses the common mathematical properties. The conclusion is given in section 6.

## 2 A Brief History of Boolean Algebra from Boole to Peirce, Venn, and Huntington

Boolean algebra derives its name in honor of George Boole [4]. During the mid-nineteenth century, several scholars were interested in formalizing alternative algebras on logic that is different from numerical algebra (algebra of numbers) [4,7]. These included Boole, De Morgan and Hamilton. De Morgan's influence on the formalization of Boolean algebra is evident in De Morgan's *Formal Logic* and examples of *symbolic algebra* [7]. In symbolic algebra, the algebra is defined by a set of *symbols* and a set of *operations* on the symbols. The different interpretations of the algebra are derived afterwards. This notion of the symbolic algebra has influenced Boole formalization of the algebra as discussed next.

In [8], Boole discussed his *Algebra of Logic* where he referred to the combinations of symbols and operations as *signs* and then continued to break the signs into *literals* (symbols) and "*signs of operations*" on the literals such as +, –, × and "*sign of identity,* =" (borrowed from numeric algebra but with different interpretations). His interpretation of a *literal* such as *x* as "*a class of individuals to which a particular name or description is applicable*". He also referred to the class of "*nothing*" to mean "*no being*" and class of "*universe*" to mean "*all beings*". Today these represent the empty set and the universal set.

As to the operations on classes, Boole used the product *xy*, where *x* and *y* represent two classes, to mean the class of things where the descriptions (or individuals) given to *x* and *y* are simultaneously true (applicable). "*Let it further be agreed, that by the combination xy shall be represented that class of things to which the names or descriptions represented by x and y are simultaneously applicable. Thus, if x alone stands for "white things," and y for "sheep," let xy stand for "white sheep;""*. With this definition, two important properties are deduced to hold true, as is the case in numeric algebra with respect to multiplication, mainly *commutativity* and *associativity*, hold true in Boole *Algebra of Logic*.

In his writings, it was a common theme for Boole to relate his form of algebra to traditional numeric algebra and point to similarities as well as to differences. Among the differences between the two algebras he discussed were the *cancellation* property, which holds true in regular algebra but not in his symbolic algebra, and the *idempotent* property, which holds true in symbolic algebra but not in regular algebra. The cancellation property is

$$xy = xz \text{ and } x \neq 0 \text{ then } y = z \tag{1}$$

The idempotent property is

$$xx = x \tag{2}$$

In continuing with comparisons of common properties of the two algebras, Boole introduced a two-value Boolean algebra as an algebra where the idempotent property does hold true as follows. If the composition is understood as a multiplication in traditional numeric algebra then the solution to equation (2) in regular algebra is $x = 0$ and $x = 1$. Hence an algebra with two classes only "nothing" (0 in traditional algebra) and "universe" (1 in traditional algebra) will work in both algebras with the operators interpreted accordingly.

Today's Boolean algebra includes two additional operations, OR (+) and NOT ( ¯ ). Boole considered these operations in [8] where he defined the binary operator + to have a meaning over classes (operands) that have nothing in common, *mutually exclusive*. $x + y$ refers to *aggregate class* (set) composed of *x* and *y*. This was done in analogy to regular algebra, the number of elements in the aggregate set is the sum of the elements in each class. This definition has changed since then and is replaced by the *inclusive* OR as suggested by Peirce and Venn [16,17]. As to the NOT operator, Boole used the notation $1 - x$ to mean $\overline{x}$ or x'. The use of the over bar, instead of $1 - x$, was suggested by Venn to remove the ambiguity resulted from $1 - x$. Later, Shannon [5] used the prime symbol as alternative to the over bar and to $1 - x$. Boole represented $1 - x$ as "*contrary or supplementary class of objects*".

Today's Boolean algebra is such that the idempotent property holds true under + as well based on Peirce and Venn modifications [16,17]. While under Boole the term $x + x$ will have no meaning due to the constraint that the classes must be mutually exclusive (not possible when both terms (*x*) are the same), starting with Peirce and following with Venn the idempotent property holds true due to relaxing the constraint of mutual exclusion.

Boole restriction of the use of + with mutual exclusive to preserve the aggregate sum as in arithmetic sum was relaxed by Peirce where + represents a *logical relation* (logical operation). To emphasize this, Peirce changed the + operation to '+,' and defined '+,' as *"a +, b denote all the individuals contained under a and b together*". That is, what is common to a and b is taken into account once. This is similar to counting number of elements in the resulting set of the union operation.

Similar arguments were presented by Venn to the meaning of logical add, $a + b$, where he considered several interpretations [17]. Venn's contribution to symbolic logic includes Venn diagrams where he represented classes (sets) using circles and represented composition (set intersection), addition (set union) and $1 - x$ (set complement) using these diagrams.

In further developing the symbolic algebra introduced by Boole, Huntington [18] has considered an alternative important approach to further the study; it is that of *axiomatic algebra* based on a set of symbols, and a set of postulates including composition postulates. This was emphasized in Huntington paper [18], "… *show how the whole algebra, in its abstract form, may be developed from a selected set of fundamental propositions, or postulates, which shall be independent of each other, and from which all the other propositions of the algebra can be deduced by purely formal processes*."

The algebra considered does not depend on the meaning or interpretation of the symbols as intended by Boole but rather on the set of postulates placed on the compositions of the symbols. To distinguish the compositions symbols found in the postulates from the traditional symbols found in numeric algebra, Huntington used the symbols ⊕, ⊙, ⋁, and ⋀ to represent combination operators and special element symbols. In the application to Boolean algebra, the symbols represent, respectively, the logical sum, logical product, the constant 1, and constant 0.

In his paper, Huntington presented three sets of equivalent postulates. We list the first set taken from [18]. "… *we take as the fundamental concepts a class, K, with two rules of combination,* ⊕ *and* ⊙; *and as the fundamental propositions, the following ten postulates*:

Ia. *a* ⊕ *b is in the class whenever a and b are in the class.*
Ib. *a* ⊙ *b is in the class whenever a and b are in the class.*

IIa. *There is an element* $\wedge$ *such that* $a \oplus \wedge = a$ *for every element* $a$.
IIb. *There is an element* $\vee$ *such that* $a \odot \vee = a$ *for every element* $a$.
IIIa. $a \oplus b = b \oplus a$ *whenever* $a$, $b$, $a \oplus b$, $b \oplus a$ *are in the class*.
IIIb. $a \odot b = b \odot a$ *whenever* $a$, $b$, $a \odot b$, $b \odot a$ *are in the class*.
IVa. $a \oplus (b \odot c) = (a \oplus b) \odot (a \oplus c)$ whenever $a$, $b$, $c$, $a \oplus b$, $a \oplus c$, $b \odot c$, $a \oplus (b \odot c)$,
and $(a \oplus b) \odot (a \oplus c)$ are in the class.
IVb. $a \odot (b \oplus c) = (a \odot b) \oplus (a \odot c)$ whenever $a$, $b$, $c$, $a \odot b$, $a \odot c$, $b \oplus c$, $a \odot (b \oplus c)$,
and $(a \odot b) \oplus (a \odot c)$ *are in the class*.
V. *If the elements* $\wedge$ *and* $\vee$ *in postulates* IIa *and* IIb *exist and are unique, then for every element a there is an element* $\overline{a}$ *such that* $a \oplus \overline{a} = \vee$ *and* $a \odot \overline{a} = \wedge$.
VI. *There are at least two elements*, $x$ *and* $y$, *in the class such that* $x \neq y$."

Huntington used *proof by deduction* to derive the different Boolean algebra properties found in today's textbooks. He also gave an example of $\oplus$ and $\odot$ tables similar to truth tables of the logic OR and AND when the class of elements, *K*, contains 0 and 1 only (two-valued Boolean algebra).

# 3 Shannon's Contribution to Digital Design

While the algebra and its definitions as an abstract algebra was founded and studied as an abstract mathematical algebraic structure, Shannon [5] related the algebra to relays design in electrical engineering. With that, the algebra found footing in digital and computer design. We look at Shannon's contribution in this section.

Shannon represented the composition operations in Boolean algebra as switches in parallel and in series. He also used the switch structure to simplify Boolean expressions. In addition, he introduced the method of proof by *perfect induction*, used to show equality of Boolean functions. Other contributions included Shannon's *expansion*, adapted from Boole expansion with regard to his special two-value algebra. The following excerpts are taken from Boole [8].

"*Suppose that we are considering any class of things with reference to this question, viz., the relation in which its members stand as to the possession or want of a certain property x. As every individual in the proposed class either possesses or does not possess the property in question, we may divide the class into two portions, the former consisting of those individuals which possess, the later of those which do not possess, the property. ... Suppose then, ..., that the members of that portion which possess the property x, possess also a certain property u, and that these conditions united are a sufficient definitions for them. We may then represent that portion of the original class by the expression ux…. Hence the class in its totality will be represented by*

$$ux + v(1 - x);$$

*which may be considered as a general developed form for the expression of any class of objects considered with reference to the possession or want of a given property x*".

Note that based Boole definition of +, the + applies to mutually exclusive classes which is certainly the case for a two-valued algebra (the universe is $x$ and $(1 - x)$). He later states "*treat x as a quantitative symbol, susceptible only of the values* 0 *and* 1" and assigns the above equation to $f(x)$. Then he solves the equation for $x = 0$ and $x = 1$ to result in

$$f(x) = f(1)x + f(0)(1 - x)$$

Shannon used $X'$ instead of $(1 - x)$ and has expanded the above equation to several variables as

$$f(X_1, X_2, \ldots, X_n) = X_1 . f(1, X_2, \ldots, X_n) + X_1' . f(0, X_2, \ldots, X_n)$$

He then proved the equality by *perfect induction* on $X_1$ ($X_1 = 0$ and $X_1 = 1$). He then repeated the above on all variables, always expanding the right-hand side, to obtain the *canonical-sum* equation

$$f(X_1, X_2, \ldots, X_n) = f(0,0,\ldots,0) X_n' X_{n-1}' \ldots X_1' + f(0,0,\ldots,1) X_n' X_{n-1}' \ldots X_2' X_1 + \cdots + f(1,1,\ldots,1) X_n X_{n-1} \ldots X_1$$

Shannon's proofs by perfect induction were possible because of the two-valued Boolean algebra chosen with known domain.

# 4 Boolean Algebra as a Lattice

Today the definition of Boolean algebra has extended to lattice theory. A good source for lattice theory and Boolean algebra is found in [20]. The section looks at this topic. Boole restricted his discussion to: composition, the aggregate operator + and the contrary operator $(1 - x)$. Lattice theory is an extension of the concepts of partial orders defined over a set of elements and a relation, R, over the set. The contribution of this section is in presenting a condensed compiling of the concepts.

A relation, R, from a set L to itself is a subset of the Cartesian product L × L and called a relation on L (or relation over L). Elements of *R* can be written as (a, b) or a R b.

Let R be a relation on a set L. We say

a) R is *reflexive* if and only if (a, a) ∈ R, ∀ a ∈ L
b) R is *symmetric* if and only if (a, b) ∈ R then (b, a) ∈ R

c) R is *transitive* if and only if (a, b) ∈ R and (b, c) ∈ R then (a, c) ∈ R.

d) R is antisymmetric if and only if (a, b) ∈ R and (b, a) ∈ R then a = b.

A relation that is reflexive, symmetric and transitive is called an *equivalence* relation. In digital design equivalence relations are used in state minimizations as an example. A relation that is reflexive, antisymmetric and transitive is said to be a *partial ordering* relation. The set with the partial ordering relation is called a *partially ordered set* and abbreviated as poset. A partially ordered set, L, is normally represented as 2-tuple (S, ≤) where ≤ represents the partially ordered relation R on L. This is in analogy with the less than or equal to operator (≤ ) as a partial ordered relation on the set of integers. *Lattices* are special types of partially ordered set on set L where for each a ∈ L and b ∈ L, the 2-tuple (a, b) has a unique *greatest lower bound* (glb) and unique *least lower bound* (lub) discussed next.

Let ($L$, ≤) be partially ordered with a, b, x, y in L. (a) x is said to be a *lower bound* of a and b if x ≤ a and x ≤ b. (b) Similarly, y is said to be upper bound of a and b if a ≤ y and b ≤ y. (c) x is also said to be a *greatest lower bound* (glb) of a and b if and only if for any other element x' in L, where x' is a lower bound of a and b we have x' ≤ x. (d) y is also said to be a *least upper bound* (lub) of a and *b* if and only if for any other element y' in L where y' is an upper bound of a and b we have y ≤ y'. An element, g, in L is said to be a *greatest element* if and only if for every a ∈ L, a ≤ g. Similarly, and element l in L is said to be *least element* if and only if for every a ∈ L, l ≤ a.

The greatest upper bound, least upper bound, greatest element and least element are unique. For completeness we include the proof of two sample cases.

Case 1: for any two elements a, b in L their glb is unique. Assume there are two glb's x1 and x2. Since x1 and x2 are both lower bounds of a and b we have x1 ≤ x2 since x2 is the greatest lower bound of a and b. Similarly, we have x2 ≤ x1 since x1 is the greatest lower bound of a and b. By the antisymmetric property x1 = x2.

Case 2: The least element is unique. Similar logic is applied. l1 ≤ l2 since l1 is the least element (l2 ∈ L hence l1 ≤ l2). Similarly l2 ≤ l1. By the antisymmetric property we have l1 = l2.

The uniqueness of the glb, lub, least element and greatest element is used in the definition of Boolean algebra as follows. First, for any two elements a and b in L, since lub and glb are unique they can be thought of as binary operations called *join* or *sum* represented by ⋁ or + for the lub, and meet or product represented as ⋀ or ˙. Since the least element and greatest elements are unique, they are represented as 0 and 1. In terms of algebraic representation, we can represent the previous as a 5-tuple (L, ⋁, ⋀, 0, 1) where L is a set of elements, ⋁ and ⋀ are binary operations, and with 0 and 1 representing the least and greatest elements of L.

With two additional properties on elements of a lattice the definition of a Boolean algebra is established: a Boolean algebra is (L, ⋁, ⋀) *where L is a distributive and complemented lattice*. With 0 and 1, a *complemented lattice* is a lattice where for each element a ∈ L there exists an element, $\overline{a}$, called complement of a, where a ⋁ $\overline{a}$ = 1 and a ⋀ $\overline{a}$ = 0. A *distributive lattice* satisfies the properties: a ⋁ (b ⋀ c) = (a ⋁ b) ⋀ (a ⋁ c) and a ⋀ (b ⋁ c) = (a ⋀ b) ⋁ (a ⋀ c) ∀ a, b, c ∈ L.

With the above definition, we can show Huntington proposition as given in the previous section hold true. This is done in the next section.

## 5. Boolean algebra as Lattices, Huntington postulates, and Shannon's two-valued Boolean algebra

In this section we look at showing the algebra defined using lattice theory satisfies the Huntington proposition. In the verification, note that ⊕ is the join (sum) ⋁ and ⊙ the meet (product) ⋀. Note as well ⋁ is the greatest element 1 and ⋀ is the least element 0.

Verification of Ia and Ib follows since by definition the glb and lub of any two elements in L is such that both glb and lub are in L as well. This is set closure under ⋀ (⊙) and ⋁ (⊕). Proposition IIa and IIb follow as well since for each element a ∈ L we have a ⋁ 0 = a and a ⋀ 1 = a . IIIa and IIIb for commutativity also holds true since the glb of (a,b) is the same as the glb of (b,a) and the lub of (a,b) is the same as the lub of (b,a). The distributive properties holds by definition of the Boolean algebra above (distributed lattice). For postulates VI there at least two elements in the class; we assume the minimum number of elements in L is two corresponding to the least element (0) and greatest element (1).

It is remaining to show the existence of the complements and the uniqueness of the complements. The existence of complements follows from complemented lattices. The uniqueness of complements follows.

Assume given an element a ∈ L with two complements x and y. Since each is a complement of a, we have

$$\begin{aligned} x &= x \wedge 1 = x \wedge (a \vee y) \\ &= (x \wedge a) \vee (x \wedge y) \\ &= 0 \vee (x \wedge y) \\ &= x \wedge y = y \wedge x \end{aligned}$$

Now starting with y we have

$$\begin{aligned} y &= y \wedge 1 = y \wedge (a \vee x) \\ &= (y \wedge a) \vee (y \wedge x) \\ &= 0 \vee (y \wedge x) \\ &= y \wedge x \end{aligned}$$

Hence x = y and the complement is unique.

## 6 Conclusion

In this paper we surveyed the algebra, Boolean algebra, from its beginning by George Boole and De Morgan's where it did not attract initial interest to attracting much interest many years later by several scholars such as Peirce, Venn, Huntington and Shannon. We then looked at lattice theory and at Boolean algebra as a special type of a lattice, a complemented and distributive lattice. We concluded with showing that Huntington postulates can be derived from Boolean algebra as a complemented and distributive lattice. Our contribution is to provide a condensed survey of the progress of the important algebra from its beginnings starting with Boole and De Morgan's.

## References

[1] J. E. Hopcroft, R. Motwani, & J. D. Ullman, "Introduction to Automata Theory, Languages, and Computation, 3rd edition", Addison Wesley, 2007

[2] Rich, "Automata, Computability and Complexity: Theory and Applications", Prentice Hall, 2008.

[3] Sudkamp, "Languages and Machines: An Introduction to the Theory of Computer Science, 3rd edition", Addison Wesley, 2006

[4] G. Boole, G. "Mathematical Analysis of Logic", MacMillan, Barclay & MacMillan, Cambridge, 1847

[5] C. E. Shannon, "The Synthesis of Two-Terminal Switching Circuits", Bell System Technical Journal, 28 pp. 417-425, 1949.

[6] C. E. Shannon, "A Symbolic Analysis of Relay and Switching Circuits", Masters Thesis, MIT 1940.

[7] A. De Morgan, "Formal Logic: or, The Calculus of Inference, Necessary and Probable", Taylor and Walton, London, 1847.

[8] G. Boole, "An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities", Walton and Maberly, London, 1854

[9] S. Brown and Z. Vranesic, "Fundamentals of Digital Logic with VHDL Design with CD-ROM, 3rd edition", McGraw-Hill, 2009.

[10] T. Floyd, "Digital Fundamentals: A System Approach", Prentice Hall, 2013.

[11] W. Kleitz, "Digital Electronics a Practical Approach with VHDL, 9th edition", Prentice Hall, 2012.

[12] M. Mano and M. Cilette, "Digital Design, 5th edition", Prentice Hall, 2013.

[13] R. Tocci, N. Widmer and G. Moss, "Digital Systems Principles and Applications, 11th edition", Prentice Hall, 2011.

[14] J. Wakerly, "Digital Design Principles and Practices 4th edition", Prentice Hall, 2006.

[15] D. Harris and S. Harris, "Digital Design and computer Architecture", 2nd edition, Morgan Kaufman, 2013.

[16] C. S. Peirce, "On an Improvement in Boole's Calculus of Logic", Proceedings of the American Academy of Arts and Sciences, 7, pp. 250-261, 1867.

[17] J. Venn, "Symbolic Logic", MacMillan, London, 1881

[18] E. V. Huntington, "Sets of Independent Postulates for the Algebra of Logic", Transactions of the American Mathematical Society, 5:3, pp. 288-309, 1904.

[19] J. H. Barnet, "Origins of Boolean Algebra in logic Classes: George Boole, John Venn and C.S. Peirce", Mathematical Association of America, 2013.

[20] A. Doerr and Levasseur, "Applied Discrete Structures", lul publishing (lul.com), 2013.

# When the Professor Joins the Student Team: A Service-Learning Project That was More Complex Than I Had Expected!

**Donald R. Schwartz**
School of Computer Science and Mathematics
Marist College
Poughkeepsie, NY 12601

**Abstract** - *The focus of service learning is to provide opportunities for our students to gain real, practical experience while completing real, practical projects, which often become extensions of the classroom learning environment. This paper describes one such project – a project proposed by our College's Medical Technology Program. The goal of the project is to schedule a set of seven on-site lab experiences for students enrolled in the Program. The set of restrictions dictating the ordering and location of these labs created a complex scheduling algorithm which required the professor to join the student team so that the project could be completed.*

**Keywords:** Team Projects and Case Studies, Collaborative Learning, Service Learning, Projects and Software Engineering

## 1 Service Learning

One of the major goals of Service Learning (also known as Community-Engaged Learning) is to enhance the classroom learning environment by bringing in "real world" experiences for our students, while at the same time offering much-needed assistance to communities in need. We can expand the learning experience our students encounter by taking the content that is already in our coursework and extending it into community service opportunities by partnering with on-campus and community-based organizations. [1] [2] Service learning is recognized as an effective means of increasing our students' social involvement, especially in their local communities. [3][4]

There is a wide variety of Service Learning projects that nicely fit into the realm of Computer Science and Information Systems. Benefits include applied experience, because our students are often very eager to develop "real" software – something that will be used by (and of use to) various community partners/clients; professionalism, since these experiences require time management skills, presentation skills and client interviews; teamwork, especially since our projects are all developed by student teams; and vocational exploration, since projects such as ours allow students to explore how the skill sets they are mastering can be used beyond the classroom and beyond their future professions. It is widely recognized that, in order for Service Learning to be as effective as possible, it must enhance the learning experience our students gain in the classroom. [5] Studies have demonstrated that students who work to meet a clearly-defined client need often have a deeper understanding of the issues their clients are addressing. [6]

It has long been my practice to incorporate Service Learning into my Software Engineering and Software Development courses, because these courses typically include semester-long projects that are to be completed by teams of students. [7][8][9] My most-recent foray into Service Learning incorporated several different projects. [10] This paper focuses on one of those projects.

## 2 The Project

The Medical Technology Program, offered within our College's School of Science, is a nationally-accredited endeavor that includes hospitals and clinical laboratories in our region of the state. Students in this program, as part of the requirements for a B.S. degree, must complete a 23-week session of active, on-site clinical laboratory experiences (rotations). These rotations cover the following general areas:

- Hematology (5 weeks)
- Blood Bank (5 weeks)
- Clinical Chemistry (5 weeks)
- Clinical Microbiology (6 weeks)
- Urinalysis (1 week)
- Phlebotomy (1 week Out-patient followed immediately by 3 weeks In-patient)

The ordering of the rotations is not prescribed, with one exception: Inpatient Phlebotomy must immediately follow Outpatient Phlebotomy (at the same site). Students participate in only one rotation at a time, again with one exception: Inpatient Phlebotomy overlaps with any other rotation (at the same site as that student's Inpatient Phlebotomy site). This overlap means that students complete 26 "weeks" of study in a 23-week session.

There are currently seven hospitals and clinical laboratories (sites) that participate in this cooperative effort. Most (but not all) sites offer every rotation throughout the 23 week schedule. Most sites can accommodate only one student per rotation during any given week; however, a few sites can accommodate two students for selected rotations. Not every site offers every rotation. Some sites offer every rotation, but cannot offer some rotations at the same time as other rotations (e.g. at site X, the Blood Bank rotation cannot be offered during the same weeks as the Hematology rotation). These restrictions, of course, define the requirements for the allowable possible student schedules.

In the past, the scheduling of students and lab sites was completed on paper by hand. Directors used colored pencils, post-it flags, and margin notes as they attempted to schedule up to 24 students per session. It is not difficult to envision the numerous, time-consuming challenges encountered each time the schedule was due. For years, these Directors had been searching for an automated means of producing schedules for students and sites.

It is my normal practice to put out a Call for Projects well in advance of those semesters during which I offer one or more service-learning based Software Engineering/Development courses. During the most-recent offering, the Directors proposed this project.

Our system allows for three levels of users: Student, Hospital, and Admin. Students may update their personal information, view their own schedule (Figure 1) and get directions to the various hospitals. Hospitals may update their information and obtain rosters and schedules for their labs. (Figure 2) Admins can do "everything", including adding and deleting users (Figure3), hospitals (Figure 4) and labs (Figure 5). Admins also have access to all scheduling activities, including having the system generate the master schedule.

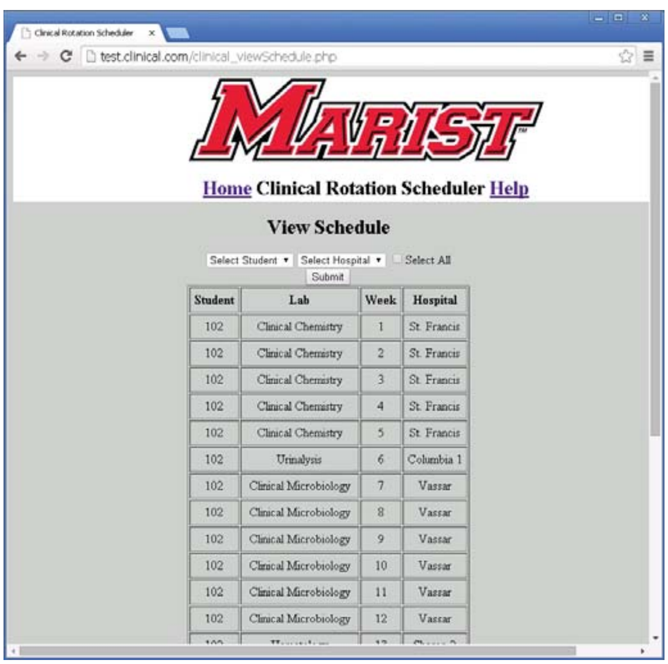



Figure 1: A Particular Student's Schedule

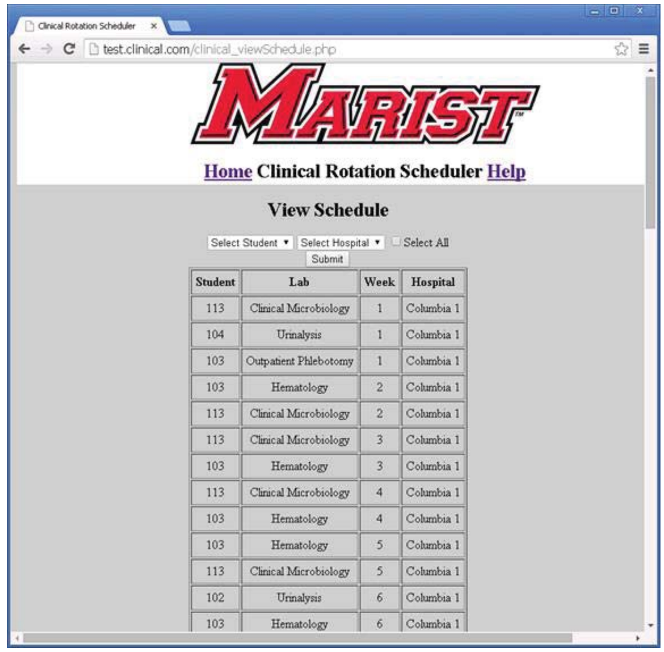



Figure 2: A Hospital's Roster of Students in Labs

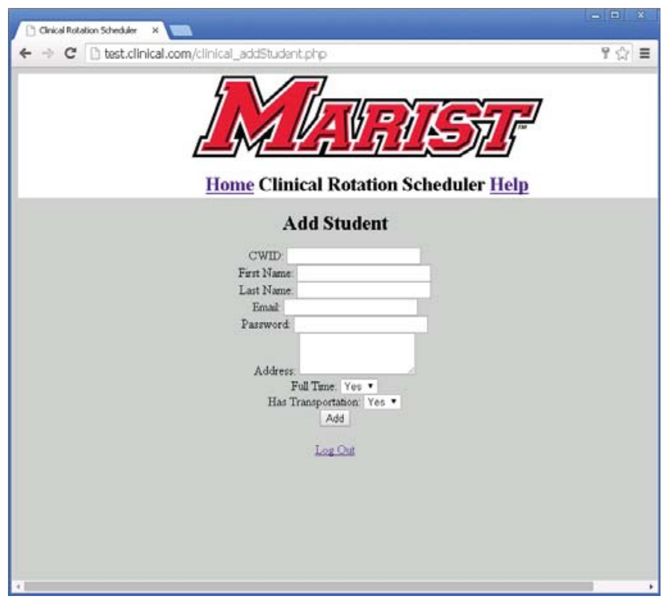



Figure 3: Adding a Student User to the System

I must admit that I was rather naïve when I first read the proposal. In the back of my mind, I envisioned this as a straight-forward recursive exercise, one which simply recursively scheduled every possible rotation at every possible site for every possible student. Once a solution was created, the algorithm would stop and display the solution. What a perfect example of using a recursive algorithm in the "real world" – ideal, in fact, for students taking Software Development II in their third semester of Computer Science and/or Information Systems! However, we soon discovered that this was not the case – not the case at all.

There were a host of complicating factors:

- each particular student's labs could begin and end during almost any week of the schedule

- the Outpatient Phlebotomy lab must precede the Inpatient Phlebotomy lab at the same site, and the Inpatient Phlebotomy lab would overlap with whatever (available) lab followed it, which meant that these three labs had to be scheduled at the same site, in the precise order, during those specific weeks

- that overlap meant that the algorithm had to find a subsequent lab that
  - the student had not already completed
  - was offered at the current site
  - was not already scheduled during the specified weeks

This required that the algorithm include a look-ahead feature that ensured that these complicating factors were successfully addressed. This was certainly significantly more involved that writing a simple recursive piece of code! By the time we discovered the level of complexity of the algorithm, and given the fact that this was an application that was truly needed by the Medical Technology Program, I quickly realized that it was time for the professor to become a member of the team. While the rest of the team worked on other parts of the project, I took on the scheduling algorithm and came up with the following approach. (Since the students were just in their third semester of studies, I tried to develop a solution that used only the data structures to which they had already been exposed. This presented yet another challenge, but one that was relatively successfully addressed.)

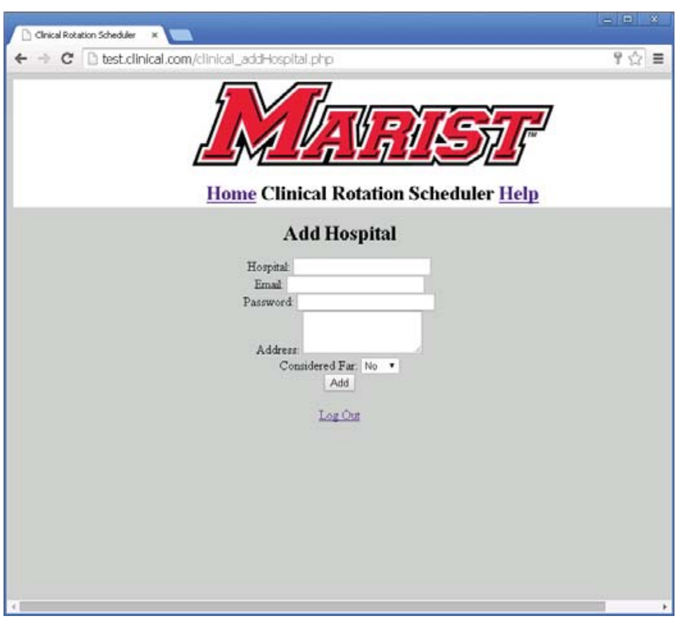



Figure 4: Adding a Hospital

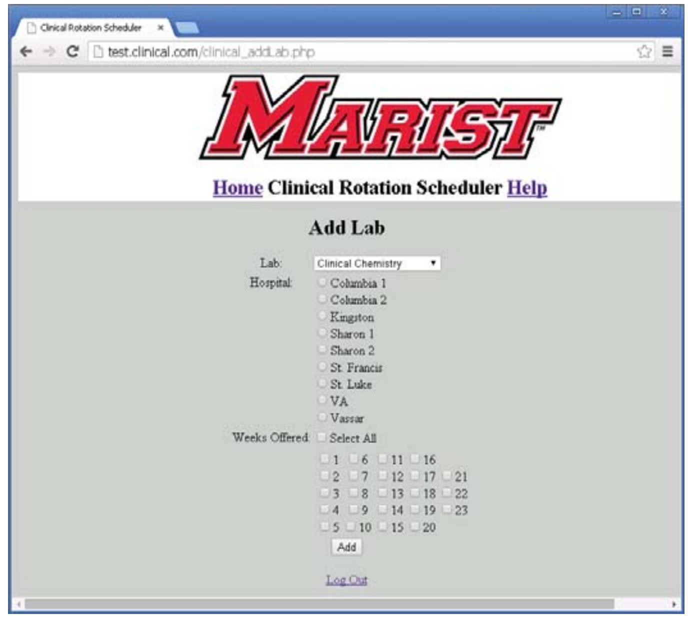



Figure 5: Adding a Lab

## 3 The Approach

Our implementation provides a means by which Directors input information about each student who is enrolled in the current session (Figure 3), along with the current session's available lab sites (Figure 4), the labs that are offered at each site, and the weeks during which those labs are offered at those sites (Figure 5). Using this input, we create a list of Students and a queue of Lab objects, each of which contains the following data:

- Site ID
- Lab ID
- Lab duration (in weeks)
- Week number (1 for the first week of the session, 2 for the second, etc.)

Here, we take advantage of the fact that the Inpatient Phlebotomy lab (three week duration) must immediately follow the Outpatient Phlebotomy lab (one week duration) at the same site – we simply combine those into a single four-week Outpatient Phlebotomy lab.

We randomize the ordering of the Lab objects within the queue, so that different overall schedules can be more-easily generated. We then populate a toBeScheduled list, which includes all students. Then, for each week of the 23-week session, for each student who doesn't currently have a lab scheduled for that week, we search through the queue for an available lab, L, one that the current student has not already completed and that is available at that same site for the next D weeks, where D is the duration of that lab.

```
From Admin input, generate all Lab objects and add them to the Queue
Add all students to the toBeScheduled list
Set the Stuck list to empty
done = false
while (not done)
    for each week of the session
        for each student on toBeScheduled who isn't already scheduled for this week
            find an available lab that isn't already on this student's schedule
                if such a lab exists
                    if that lab happens to be Outpatient Phlebotomy
                        find an additional available lab at the same site
                        that starts next week and that is not already on
                        this student's schedule

                        if such a lab exists
                            add Outpatient Phlebotomy and the additional lab
                            to this student's schedule

                        else return Outpatient Phlebotomy to the queue
                             and try again for this student

                    else add that lab to this student's schedule
                else return every lab on this student's schedule to the queue,
                     fill her schedule with placeholders and
                     add student to Stuck list
        end-for-each-student
    end-for-each-week

    if StuckList is empty
       done = true
    else for each student on StuckList
            clear out the placeholders from her schedule
            move student back to the toBeScheduled list
end-while-not-done
Output schedules
```

Figure 6: The Scheduling Algorithm

Those requirements are not very difficult to meet. For Student S, for a given lab, L, we check to make sure that there are Lab objects for site H, lab L, weeks W through [W + (duration of L)], removing those Lab objects from the queue and adding that information to student S's schedule. However, if that available lab happens to be Outpatient Phlebotomy, we must utilize a look-ahead feature. (Remember that Outpatient Phlebotomy is immediately followed by a three-week Inpatient Phlebotomy lab, which can overlap with the "next" lab a student takes.) We must find an additional available lab (at the same site, H) that starts during the "next" week (W + 1). For example, suppose we are trying to schedule student S (who has not yet completed Outpatient Phlebotomy) during week W, and we have found that Outpatient Phlebotomy is available at site H during weeks W through W+3 (since Outpatient Phlebotomy is treated as a four-week lab). Before we can add this lab to student S's schedule, we must also find another lab, L2, that student S has not yet completed and that is available at site H during weeks W+1 through [W+ (duration of L2)]. If such a lab L2 exists, then we add Outpatient Phlebotomy and lab L2 to student S's schedule for weeks W through L2's duration plus one. If we cannot find such a lab L2, then we cannot schedule student S for Outpatient Phlebotomy during week W and must therefore return those Lab objects (Outpatient Phlebotomy and L2) to the queue. We then try to find another lab (at any site) for student S to take during week W.

Ideally, as we work through the loops in our algorithm, we will come up with a schedule for every student. If not, we need to find a way to keep trying to schedule the other students. We've included a counter that tracks the number of times we've attempted to schedule a lab for a student during a particular week. If no lab can be found, we return all of that student's scheduled labs to the queue, then remove that student from attempting to schedule any more labs. (This is accomplished by filling out that student's schedule with placeholder labs and adding that student's ID to a Stuck list.) When no more students remain in the toBeScheduled list, we then check the Stuck list and for every student there, we remove all of their placeholder labs and add the students to the toBeScheduled list and start their schedule search again. If we still can't find a schedule, then we output a failure message and run the entire algorithm (for every student) again.

Once a schedule is created, we output schedules for each individual student (Figure 1) and rosters for each rotation at each site (Figure 2).

The overall algorithm is given in Figure 6.

## 4 Future Work

Our clients for this project were very pleased with the application that we presented to them. They are quite eager to use the product during the next round of clinical rotation offerings. As a matter of fact, we've already received inquiries from other programs with similar needs, wondering whether this application will work for them, too. Anticipating that this might be the case, we developed a solution that was as generic as possible, so that it might be readily adaptable to other uses.

After the application is tested in a "real" working situation, but before we make it more widely available, there is at least one area that must be addressed more fully. As was previously mentioned, this project was one of several service-learning projects we completed in Software Development II, which is the third course our majors take. It should be apparent from the screenshots included in this paper that the user interface leaves a lot to be desired. Fortunately, I also teach Systems Design, which our majors normally complete during their fourth semester. This course includes a major unit on User Interface Design. This project should make an ideal assignment for "improving an existing user interface", especially when given to a class that includes none of the students who helped develop the original project. I am very eager to see what improvements the new students develop!

## 5 References

[1] Kaye, Cathryn Berger. The Complete Guide to Service-Learning. Free Spirit Publishing. Minneapolis, MN, 2004.

[2] Tinkler, Alan, Barri Tinkler, Ethan Hausman, and Gabriella Tufo Strouse. "Key Elements of Effective Service-Learning Partnerships from the Perspective of Community Partners", Partnerships: A Journal of Service-Learning & Civic Engagement, Vol. 5, No. 2, Fall 2014, pp. 137-152.

[3] Schatteman, Alicia M. "Academics Meets Action: Community Engagement Motivations, Benefits, and Constraints", Journal of Community Engagement and Higher Education, Volume 6, Number 1 (2014), pp. 17-30.

[4] Kilgo, Cindy Ann, Kira Pasquesi, Jessica Kathryn, Ezell Sheets, Ernest T. Pascarella, "The Estimated Effects of Participation in Service-Learning on Liberal Arts Outcomes", International Journal of Research on Service-Learning and Community Engagement, Volume 2, Issue 1, 2014, pp. 18-31.

[5] Nejmeh, Brian A., editor. Service-Learning in the Computer and Information Sciences: Practical Applications in Engineering Education. Wiley-IEEE Press, June 2012, 598 pages.

[6] Eyler, Janet. "Reflection: Linking Service and Learning—Linking Students and Communities", Journal of Social Issues, Vol. 58, No. 3, 2002, pp. 517-534.

[7] Schwartz, Donald R., "The Benefits (and Drawbacks!) of "Real World" Software Engineering Semester Projects: A Case Study", Proceedings of the 2009 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS-09), Las Vegas, July 2009.

[8] Schwartz, Donald R., "Service Learning and Software Engineering: Managing 'Real World' Group Projects", Proceedings of the 2007 International Conference on Software Engineering and Data Engineering (SEDE-07), Las Vegas, July 2007.

[9] Schwartz, Donald R., "Managing and Assessing Software Engineering Group Projects", Proceedings of the 2005 American Society for Engineering Education Annual Conference and Exposition, June 2005.

[10] Schwartz, Donald R., "Offering Service Learning Projects in a New Environment", Proceedings of the 2014 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS-14), Las Vegas, July 2014.

# A Multi-Disciplinary Mobile Applications Project Course at the Graduate Level

**Jonathan Rose[1], Braiden Brousseau[1], and Alexandra Makos[2]**
[1]Department of Electrical and Computer Engineering, University of Toronto, Ontario, Canada
[2]Ontario Institute for Studies in Education, University of Toronto, Ontario, Canada

**Abstract**— *We describe five years of experience and learning with a graduate project course that brings together students from many disciplines – health, education, music, science, etc. – with graduate-level programmers from Computer Engineering and Computer Science. The goal of the course is to create a prototype of a mobile application in the field of the non-programmer. The course provides experience in cross-discplinary ideation, interaction and communication, and has resulted in a number of exciting ideas. This paper describes the structure of the course as it has evolved, with commentary on what works and does not, including assignments, group forming, and project milestones. We then describe a few of the specific project outcomes, give a general sense of the remainder, and describe issues and future directions for the course.*

**Keywords:** Projects, Multi-Disciplinary, Mobile, Wearable

## 1. Introduction

In 2007, mobile technology took a revolutionary leap with the introduction of the iPhone, and the subsequent opening of the programmability of that phone to anyone who could program a computer. Mobile and wearable technology bring together a powerful internet-connected computer with novel sensors and output devices that have enabled incredibly creative new applications in almost every field of endeavour, giving rise to perhaps one of the greatest surges of creativity in human history. The ubiquity of these smart mobile devices means that many people are inspired to conceive their own ideas for new applications, but only those skilled in the art of programming can create them. However, people without programming skills do have great expertise and ideas that could drive new applications if they were somehow enabled to make them. Conversely, the set of ideas from those in the programming disciplines is limited by their area of expertise, which is often purely technical.

Our notion, begun in 2010, was to bring together students at the graduate level in many disciplines, together with graduate-level programmers to work collaboratively to create prototype applications in the field of the non-programmer. Our assumption was that students at the graduate level in various disciplines would bring a level of expertise understanding and insights from their field, to drive new applications that would either aid them in their research, or do something novel in their general field. We also assumed that graduate-level programmers would have the skills and drive to create a high-quality prototype in the short time frame of a single-semester course.

We also saw the course as an experiment in inter-disciplinary project work that would benefit students of both types by exposing everyone to issues in inter-disciplinary communciation and engineering project work.

The results have been delightful and inspiring, rising to the level of pointing out new directions for research. There have been students from many disciplines taking the course - medicine, music, psychology, anthropology, biomedical engineering, education and many more - together with many students from computer engineering and computer science. Universities have a unique ability to create experiences such as this; one reason for writing this paper is to describe how relatively easy it is mount this kind of course, and to encourage others to do so.

This paper is organized as follows: the next section describes the two fundamental natures of students taking the course, their requirements, and the implicit bargain they make in undertaking the project. The subsequent sections describe the structure of the course - the project structure, lecture content, and assignments. We then describe several specific projects as example outcomes, and give a general account of the larger number of projects over the five years. Finally, we describe several issues and our plans for dealing with them in future iterations of the course.

## 2. Student Types and the Bargain

The fundamental assumption of the course is that there are two types of students: those who bring programming skills and experience (called, quite naturally, *Programmers*) and those who bring expertise and knowledge from another discipline. The latter we call *Appers*, rather than the more obvious but negatively connotated 'non-programmers'. Early on, by the second lecture, students must declare whether they are a Programmer or an Apper, as they will follow two different paths. It is also possible for a student to declare as both, if that they possess the requirements of both.

We require that a Programmer has fairly extensive course-based and project-based software experience. Our experience has found that those with limited programming backgrounds (perhaps having taken just one or two introductory programming courses) are unable to both learn the mobile programming environment and make a meaningful contribution to the project. For this reason we ask the Programmers to describe their relevant course work and project experience as part of their first assignment. The criteria for a Programmer are:

1) Two introductory programming courses, including one that describes data structures and algorithms. The latter course should not be theory-only, but include practical programming assignments.
2) Several courses in the areas of operating systems, graphics, databases or software engineering. These kinds of courses often have intensive programming experiences in which students have to work on large amounts of code in a short amount of time - crucibles for good programming experience.
3) Project experience in which the Programmer has personally created a relatively large amount of code in a functional programming language - preferably 2000 lines or more. This experience can be in prior course projects or capstone design projects, or in internship/work experiences.

If the student's qualifications appear to be insufficient, then we have a conversation to establish a clearer understanding of their background. If the background is not strong enough, we ask the student to leave the course.

A key requirement of the Appers is that they bring expertise from a specific discipline. An easy example would be a medical doctor taking a graduate degree, who clearly brings the expertise of medical practice and perhaps even a specialty. Another example is a graduate student in an education research program who brings expertise in the understanding how people learn. There are many more good examples from graduate students in music, psychology, and civil engineering; several of these will be describe in the examples section below. We have allowed Appers in one graduate field to make a 'claim' to expertise in another field - for example, an Electrical Engineer who had spent a good part of his life composing music wanted to drive an application in the music composition field. Upon discussion, he was able to substantiate this claim with sufficient experience. On occasion, Appers arrive in the course, but are unable to make a clear claim to a specific area of expertise. Our experience has shown that these students will both have trouble attracting programming partners (an important part of the course process described below), and that if they do, that the resulting proposal and project is weak and un-directed; we now move to suggest that these students leave the course. (In both cases, asking students to leave is not a pleasant experience for them, but it is far better than the struggles we have seen in previous years when they stay).

The expertise of the Apper is an essential element of the course, as is the high-level programming capability of the Programmer(s). We view the course as a *bargain* between these two parties - the Apper brings multiple years of expertise in a field providing knowledge and insight that lay people do not have, and sometimes access to sophisticated facilites that can aid the project. An example of the latter is an Apper in rehabilitation sciences who has access to a rehabilitation measurement lab and equipment that could be used to measure the accuracy of a smartphone-based instrument. The Programmers give, in return, their labour, programming skill and technical insights as to what is possible, feasible and achievable.

This gives rise to one of the key rules of the project: the application must be within the field of the Apper, and not something else for which they have an idea, *but no expertise*. This rule is rather important, as some number of Apper candidates arrive in the course with ideas for applications quite outside of their field – perhaps wanting to make a game or some kind of social networking app unrelated to their field. While these might be good ideas, without the backing expertise there is less likliehood of novelty and authoritative insights.

## 3. Project Structure

The central goal of the course is to form an interdisciplinary team and to create a prototype of a novel mobile application in the field of the Apper. The timeline of the 14 week course is given in Table 1; this section will describe the core components of each activity.

| Week(s) | Project Activity |
|---|---|
| 1-2 | Introduction & Team forming |
| 3 | Idea Forming |
| 4 | Approval-in-Principle |
| 5 | Written Proposal/Plan Due |
| 6 | Proposal/Plan Presentations (extra lecture) |
| 7 | Reading Week - no class, work though! |
| 8 | User Experience Lecture |
| 9-10 | Spiral 2 Presentation & Demo |
| 11-12 | Spiral 4 Presentation & Demo |
| 13-14 | Final Presentation & Demo |

Table 1: Project Timeline

### 3.1 Team Forming and Vetting of Students

Perhaps the most difficult and chaotic part of the course is the process of forming teams. After some experimentation, we have landed on specifying that a team should consist of one Apper and two Programmers. It makes sense to have one domain expert in a project; on occasion a sginle student has served as both Apper and Programmer. We have found that the typical amount of programming work is extensive and likely too much for one person, and hence the two Programmers. Two programmers also serve to help each other learn any new material in the event that one gets stuck. The scope of the projects involves such a large range of the computer engineering stack that multiple programmers can complement each other's specific expertise in the various areas such as web/server, UI, optimization, and computer vision. On the other hand group dynamics become more difficult with three Programmers, and so we allow this only under exceptional circumstances (typically if another group disintegrates and a Programmer is looking for a home, we will assign that person to a group with particularly ambitious programming needs).

One explicit rule of the course is that a student can only stay in the course if they successfully become part of a team. This requires a 2:1 ratio of Programmers to Appers, which can only be in-directly enforced by the rule; however our experience has been that this ratio has always roughly been present, even at the beginning of the course. Previous research suggests that allowing students to choose their own teams results in choices that are based on a mixture of predictions of success (i.e. choosing people who will help them succeed) and finding partners who are self-similar [1], the latter limiting the former. There is one extra constraint in our projects that influences us to allow student-led group selection: we want everyone to be excited by the topic of the project they are working on, and so want the team forming to include this freedom of choice. So, a Programmer who is keen on doing a sports-oriented application can freely gravitate towards an Apper in Kiniesology, rather than someone in the Education program.

For students to find out about potential partners, the very first assignment for both Programmers and Appers requires them to both describe themselves in writing, and to make a video - both of which are posted on a course-accessible website. In both, the Programmers are asked to describe their background, their software project experience, and what kind of project they are interested in. This is used by the instructors and TAs to ensure that their software capability is sufficient to withstand the stress of the course, as their part of the 'bargain' described in Section 2. It is also used by the Appers to evaluate the capabilities and communication skills of potential partners.

In the document and video, the Appers are asked to define their area of expertise, and to float ideas of what they have been thinking about as potential applications. We assume that every Apper has already pondered this question, as that is the reason they were taking the course! The instructor and TAs use this information to determine if there is a clear definition of expertise (which is the Apper's side of the bargain).

Since a large part of the course consists of communication that is both written and verbal, we feel that having students displays their communication capability early on is important.

In the 2015 year we took a new approach to forming teams, encouraging Programmers to find a compatible programming partner first (who they'd like to work with, and who has similar areas of interest when seeking a topic and Apper). After forming a pair, the team would interact with Appers to find their third partner. We provided an extra week for this pairing step to take place.

In previous years we have finalized the team-forming process by having a separate class get-together, in the evening to close on finding a team. At this event we have the Appers give one minute summaries of their area/ideas, verbally, and then have the Programmers informally talk to potential partners. In the current year with the extra time available, many of the groups were formed prior to this point in time, and so the extra time was used to have the un-attached Appers make 3-minute pitches to the un-attached Programmers. This latter process was possible as there were only a 5 un-attached Appers at that point, and three pairs of Programmers. This did mean that two Appers were unable to continue in the course.

In addition to the viewing of the written documents and video (which are due within the first week of the course), several other opportunities are given for the students to get to know each other. The last parts of the first and second lectures, are used for the Appers and Programmers to introduce themselves and talk about their background and ideas. We use these very informal statements as opportunities for class-wide discussion of ideas, leading into the idea creation phase, next.

## 3.2 Idea Creation and Approval-in-Principle

As the teams are forming, and once they are formed, there is much informal discussion of project topics. As discussed above, the topic must be in the field of the Apper. As this is a graduate course, the scope can include mobile applications that might help the Apper in their graduate-level research (for example help collect data among patients in a novel way) *or* be something that would augment their field more generally (such as a teaching aid in education). The Apps may well be something that could be commercialized, but this possibility of a research/field mandate sets the scope to be super set of those applications that are of commercial value.

Teams are encouraged to propose ideas via email to the instructor and TAs to get rapid feedback. The projects proposed must be of sufficient technical depth; this is sometimes a function of the Programmers' capabilities. This rule prevents simple information-based apps that would be the mobile equivalent of a document. There must be something that has some learning/challenge/effort at the graduate level of the Programmers. The instructor reserves the right of approval on this basis, and it is done informally via email. Some of the 'ideation' is driven by the content of the first four lectures, which describe the capabilities of mobile and wearable devices, and give examples using those capabilities, as described below in Section 4. The key milestone is to received 'approval-in-principle' on the topic and rough scope, and is due in week four of the course.

At this point (and the next) in the process, it is crucial to spot and correct especially fuzzy thinking - it is not uncommon for there to be an unfocused idea that is too broad and lacking in clarity. If left uncorrected, the final project has typically been poor. Sometimes, the act of enforcing focus reveals an inability to do so, and it is better for the student to leave the course at this point. Other times students given clear feedback on the need for focus and clarity return with far superior ideas-in-principle.

### 3.3 Proposal/Plan

Given approval-in-principle, the team is now asked to write a short proposal and plan of app structure/work. A written proposal/plan is due in week five, and is presented in week six.

The proposal reiterates what the project is, and its motivation. The plan gives a rough design of the overall structure of the work, and a set of milestones to be achieved of the ensuing six weeks. For the structure, we encourage students to present a block diagram of the entire functionality of the App and any connected hardware or software. The Appers are typically unfamiliar with the notion of a high-level block diagram; it is also surprising that a subset of the Programmers are also unfamiliar. Hence we review the concept, and illustrate it with a discussion on what would be involved in a hypothetical application's block diagram.

One of the most exciting parts of the course comes in the ideation phase: because many Apps are connected to us as humans (being portable), we can all fairly easily imagine ourselves using a proposed application, and to ponder whether it will work, and what else it might do. We call this process 'living within' the proposed Application. That is, we mentally place ourselves within the context of using the App, and see if there are new and related ideas that could bring more capabilty/functionality. To teach this notion, we have the class as a unit *live* in a particular example; an App that measures the ability of a person to balance, as a metric of sobriety. We then ask how this functionality might be used - by police, bartenders, spouses, etc. and how that might work. Over the years of the course, some wonderful inspirations have resulted from this process, some of which are described in Section 6.

The proposal presentation is limited to six minutes - every presentation in the course has a similarly short time limit; not only to allow the acommodation of up to 20 projects, but to ensure brevity and clarity. The class as a whole along with the TAs and instructors respond by providing feedback and asking questions. We also provide written feedback along with the grade for the proposal/plan. This point in time is also key for providing projects with a lack of focus or incorrect scope the needed guidance.

For the plan, we advocate making use of the general engineering method known as the *agile* or *incremental* or *spiral* method: *the idea is to make the simplest evocative prototype as soon as possible, and then iterate on it, adding improvements.* The weeks from the proposal/plan week are numbered starting from week 0 in the plan. We suggest selecting clear target of functionality for a prototype be ready as soon as week 1 is over, and call that the 'Spiral 1' target. Subsequent weeks are Spiral 2, 3 and so on. As such, the plan isn't required to have much detail as the subsequent goals are more readily identified as time progresses, as per the spiral/agile method [2].

An important part of the proposal presentation is from the Apper, who typically gives the introduction and motivation sections; we also ask the Apper to describe what their role will be in the project execution, and what else they will bring. Two examples of this are the use of laboratory facilities that can help in measurements the project/App makes, or the application of their expertise in music analysis.

We believe that one of the key reasons for the success of the projects in this course is the requirement that students present, in the class, their progress and a demonstration of the partially-working App, as of the Spiral 2 and Spiral 4 deadlines, prior to the final presentation and demo. This forces work to happen at regular pace - hard deadlines and a public presentation are very focusing!

Since it takes roughly four hours to do the proposal/plans for 15-20 projects, and our lecture is only 2 hours/week, we have added in an extra lecture for the proposal week to launch everyone at the same time.

Finally, at the proposal stage, we ask the team to give a name to their project/app. This name becomes the label by which we refer to the project, and very quickly becomes the identity of the team.

### 3.4 Execution and Demonstrations

The final six weeks of the course are devoted to the work to build the prototype, and the series of interim and then final presentations/demos. The actual work goes on outside of class, and the students must meet regularly to make that happen. As we typically have on the order of 15-20 projects, it takes one entire lecture to get through ten of these presentations, including time for feedback and questions. The Spiral 2 and 4 presentations are required to be 5 minutes long, and are similar in structure: a quick reprise of goal, a description of the progress made, a demonstration, and then a plan for what the next demo will bring. The class and instructors respond with questions, feedback and suggestions. These cover everthing from helping to solve technical challenges (or warning of some to come), re-directing the goals in the face of new issues or exciting discoveries, as well as feedback and suggestions on the quality of the presentation itself. The presentation is graded and given additional written feedback.

The final presentation and demonstration happens in the last two weeks. The students are given a longer presentation time (8 minutes), and asked to make a presentation that is self-contained. You can view the videos of many of the final presentations from 2013 [3], 2014 [4] and 2015 [5].

## 4. Lecture Content

The lecture content is confined largely to the first five lectures. Some of the content is the project organizational structure, described above. The key material is a description of the capabilities of mobile technology, and examples of their use in applications. These applications include prior projects in the course, and interesting and innovative applications from the literature and commercial world. The series of lectures can be found by going to the link in this reference [6], and clicking on the ‘Content’ link.

The stunning capabilities of mobile technology arise from the miniaturization of integrated circuit and sensor technology - the one portable device contains a powerful computer, a high-speed connection to the internet, a high-resolution screen, high-quality sound output, and a series of cheap but accurate sensors: the accelerometer, gyroscope, magnetometer, microphone, light sensor, proximity sensor, GPS and barometer. Remarkably, the sensors are sampled at rates ranging from 10 to 100 samples/second, providing very frequent measurements of acceleration, angular motion, light, magnetic fields, air pressure and more. The processor glues all of these sensors and outputs together with all of the algorithms ever developed in computing. Generally speaking, most people do not know about all of these different sensors; they may have used them inadvertently without even knowing that they exist. The details and example uses of these sensors, described in detail in Lectures 1, 2 and 3 are a key source of ideation.

There are also some extraordinary examples of the use of these sensors - for example, the digital signal processing algorithms developed in [7] are able to infer a human’s heart rate by analyzing video of a person’s face – the invisible-to-the-human-eye change in colour with each heartbeat can be detected with under 20 seconds of video. This *super-human* capability is inspiring, and we suspect there are many of these that can be used in future-looking applications.

More recently there has been a surge of smaller wearable devices that are wirelessly connected to the smartphones. Thes have sensors in smaller, more portable/wearable packages. One low-cost example is the $29 ‘Sensor Tag’ from Texas Instruments [8], which is a bluetooth-connected set of buttons, accelerometer, gyroscope, magnetometer, humidity sensor, ambient temperature sensor, and directional temperature sensor. Another example is the ‘Node’ sensor from Variable Inc., which includes the accelerometer, gyroscope and magnetometers in the base unit, and can add two other sensors either end - temperature, weather, colour, gas and many others [9]. Even smaller are the bluetooth-connected trackers that can indicate the presence or absence of anything they are attached to – for example the TrackR device [10].

## 5. Assignments

The first month of the course is taken up with two key tasks: the Programmers and Appers need to come up to speed on the mobile programming environment, and, as described above, form teams and generate ideas for projects. In the following sections we describe the assignments, which can be viewed under the Assignments link of [6].

### 5.1 Programmer Assignments

For the Programmers, this part of the course is like many other programming courses in which they learn about a new kind of programming paradigm/environment and infrastructure by doing assignments that take them from the simple basic capabilities to the complex. The challenging part is that they have to do this in one month, rather than 3 months for a typical undergraduate course. We rely on the fact that these Programmers are at the graduate level, and have experience and sophistication and so are able to pick up the material more quickly. The four assignments described below cover the basics that almost every App in the course will need:

**1. Development Environment & Simple Widgets.** This is the basic environment set-up, creating the first simplest program (‘hello world’) and making a slightly more complicated App that receives user input, and produces a few outputs. At this point the Programmers have to have chosen which environment they will use for the course - typically Android, but some choose iOS. This choice is important, as they will have to find a programming project partner who wants to work in the same environment. For learning resources we use both the online Android documentation and one of two book series [11] [12]. All the assignments give sections of the texts to read.

**2. Containers, Fragments, Select, Lists and Files.** This assignment teaches the basic methods to display lists of items - a very common attribute of many Apps - and how to store information and retrieve it from files. It also teaches how users provide various kinds of input,and conveys the basics of the view hierarchy of both Android and iOS. Students who are unfamiliar with event-driven systems will be exposed to various aspects of that concept.

**3. Location, Motion Sensors and Image Capture.** This assignment gives exposure to several of the key capabilities of modern mobile devices - determining the geopgraphic location of the phone, measuring its motion through the use of the accelerometer, and capturing images from the camera. Many applications make use of one or more sensors.

**4. Threads, Databases and Network Connections** The final assignment deals with creating a local SQL database, and spawning threads to perform tasks independent of the basic User Interface thread. These are important as many apps require databases, and often perform compute or network tasks that would slow down the user interface if they were not spawned as a separate thread.

### 5.2 Apper Assignments

The goals of the assignments for the Appers is to expose them to the capabilities of mobile technology, to explore what has already been done in mobile in their field, to give them experience in the user interface design, to be exposed to language and concepts of one essential capability/discipline of computer engineering, and to practice being creative in the mixed milieu of technology and their field. It is worth noting that in the first version of the course in 2011, we had the Appers try to learn a simple form of programming, based on a visual programming framework called Google App inventor that itself was based on the MIT Scratch environment [13]. We found that the Appers with no programming background were easily able to do very simple apps with a few buttons that cause actions. However, once the concepts of variables and loops were in play, they quickly became lost. In subsequent years we focused the Appers towards understanding capabilities of mobile devices, understanding computer concepts and creativity exercises. The four current assignments for Appers are:

**1. Connecting Your Field to the Mobile Devices Field.** Here the Appers are asked to survey the landscape of mobile applications in their own field - they are to look at descriptions of five different applications (available on an App Store or described in a paper) in their field and write a 100 word summary of it. Then, they are to acquire one of these Apps (one that they can run on their own mobile device) and to write a more extensive review of it, including its motivation, why it is interesting, how it works and suggestions on how to improve it. This is important so that the student can get a sense of the prior art, and when it comes to inventing their own ideas, not replicate what has been done.

**2. App Design Principles, Mockingbird & Practice.** One part of the project that all Appers can participate in actively, is the user interface look and feel. There exist a number of different free, online tools for both drawing and linking different screens of an App [14][15] [16] These tools are quite easy for anyone to learn, as they are similar to regular picture-drawing programs. The assigment asks the Appers to first read about some basic principles of user interface design (provided by Apple [17] and Android [18]) and to practice building simple designs using one of the above tools. Then, they are asked to create the complete user interface design and flow/links between screens for a specific application. At the same time, to get the 'creative' juices flowing, the assignment asks them to invent a new application based on a specific novel capability - in this case the ability to listen to a conversation among multiple people and determine what fraction of the conversation each person takes up.

**3. Understanding Parts of the Canvas.** The third assignment makes use of the fact that the Apper will have formed a team with two other Programmers at this point. The assignment gives a set of basic technologies/capbilities present in all modern networked computers – search, databases, signal processing, optimization, and internet communication – and asks the Apper to choose one of these. They must then spend time with their programming partners to learn from them the basics of this capability, and then to explain what they've learned in their own words. After that they are asked to do their own independent research to augment what they've learned and to write that up at a later time. The goal here is to have the Apper cross over into the technical realm to a certain depth - to understand *what* is being achieved in that realm, but not really *how* it is done, as that is typically too complicated. It also sets up a pathway of dialogue between the partners in the team. Our expectation that dialogue in the other direction (from Apper's field to the Programmers) happens naturally as part of the project.

**4. Creativity, Sensors and You.** The goal of this assignment is to stimulate the creation of ideas, connecting to some of the early lectures on capabilties of smartphones. The Appers are asked to invent ideas for interesting Apps in their field that make use of some of the smartphone sensors - the accelerometer, gyroscope, barometer, camera, light sensor, proximity sensor, humidity sensor, etc. It also asks them to consider the amount of processing that might be needed in their idea - for example, if they use a single picture, they'd need to count the number of pixels to be processed in an image, or much more for a video. The second part of the assignment suggests several new sensors that may appear in the future (a gesture sensor, an ultrasound imager, an emotion sensor, a blood pressure sensor, and a brain activity sensor) and asks the student to come up with ideas in their field making use of these.

## 6. Outcomes

The course has been running for five years, with the fifth year in flight as this paper is written. In total, including this year, there will have been 95 completed projects with 76 Appers and 187 Programmers. The projects have been in many of the areas that the University covers in its graduate program, including Aerospace, Anthropology, Biomedical Engineering, Drama, Education, General Medicine, Industrial Engineering, Library Science, Music, Museum Studies, Nusring, Pharmacy, Physiotherapy, Psychology, Rehabilitation Science, and Surgery. It has been thrilling to watch the teams come together and create novel and interesting applications in these fields. It is clear that, in the best projects, there is a great deal of inter-disciplinary learning as the Programmers become familiar with the basic concepts and language of their Apper's discipline. For many of the Appers it is their first experience in a serious engineering project, and their exposure to the concepts of agile/spiral development, and the complexity of software has been a formative experience for them. The experience in the course will enable them to more easily collaborate with engineering development in the future, and possibly make them more likely to realize their ideas in collaboration with an engineering team. It is also clear, from the increase in quality of the presentations over the four opportunities in the course, that the strong emphasis on communication engenders significant improvements in many of the student's ability to communicate.

To illuminate some of the outcomes, we describe three of the projects in more detail below, and provide a table summary of several more after that.

### 6.1 iAnkle

The goal of the iAnkle project [19] was to help a person with an injured ankle (either a break or sprain) recover. This is done by both measuring the stability of the ankle and prescribing exercises to progressively improve the stability. The Apper is a licensed Physiotherapist who was completing a Master's degree in public health. The basic idea, conceived by the Apper, was to use the accelerometer in a phone (tucked in a sock) to measure how much the patient would 'wobble' when doing various kinds of balance exercises, such as standing on one foot. The prototype both quantified the wobble through measurement, and attempted to prescribe successively harder exercises as progress was made. The measurement with the phone's accelerometer was a proxy for far more expensive force plates that are used in experimental physiotherapists labs, and the Apper, with the help of the Programmers, was able to bring that to a low-cost prototype that has the potential to be used by anyone with a smartphone. This very exciting application continues as a research project, with the Apper heavily involved.

### 6.2 Baton

The Apper in the Baton project [20] is an experienced high school teacher who was taking a Master's degree in Education. He sought to find a better pathway of communication between a teacher and a class of students during a class discussion. Rather than simply putting up one's hand to make a contribution, the students were to use a smartphone application to indicate both an interest in communicating, and also the nature of the contribution - either building on a previous comment, contradicting it, or moving to a different topic. The teacher's receiving application (connected through a server) would be notified of who was involved, and how long they had been waiting to contribute. If you watch the video available under the link [20] you'll see the Apper and Programmers give an eloquent demonstration of the concept.

### 6.3 Mindful Me

The Apper in the Mindful Me project [21] was a Ph.D. student in Psychology, with a focus on the treatment of addiction (to alcohol, drugs, tobacco, etc.). A standard way that therapists ask a patient to help themselves is to ask them to write into a journal a description of their cravings – when they happen and under what circumstances. This is used in a process of reflection and understanding, part of the process of resisting a relapse. The Apper's idea was to make this journal into a smartphone application, to gain ease of use and more privacy (as no-one would ask someone typing on a smartphone what they are doing, whereas they might ask about writing into a journal). A phone could also more easily record time, and location (using the GPS) and possibly other things about the state of the patient. During the inital proposal and spiral 2 presentations a very interesting thing happened - it became clear that a phone could not only act as a recorder of events, it could become an actor that actually influenced events. Consider an alcoholic who appears to be moving towards his favourite bar (as monitored by the GPS) or a drug addict heading towards the place where she purchases her favourite drug. A phone could notice this and try to intervene! This raised the very interesting question of what such an intervention might be – such as the phone playing a song, speaking a mindfulness text, or calling a friend or sponsor. This is one example of how 'living in' an App brought forth an exciting idea.

### 6.4 Summary of Selected Projects

Table 2 gives a short list of several other projects - their name, year, and very short description. Longer descriptions of each of these (a final written report and a video presentation) can be found by looking under each year's website archive, which can all be found at the bottom of the page of reference [6].

| Year | Name | Description |
|---|---|---|
| 2011 | BrainEX | Brain exercise to combat dementia |
| 2011 | Wound Capture | Record wound assessment and treatment |
| 2011 | Whimper | World noise mapping project |
| 2012 | EYEdentify | Game helps autistic children learn emotions |
| 2012 | DriveMod | Monitor and measure car driving quality |
| 2012 | SurgicalBlackBox | Surgery video & data review/annotation |
| 2013 | Mobile Stage | Augmented reality interactive theatre |
| 2013 | LunchTime | Helping children learn to tell time |
| 2013 | SnapNDose | Proper dosing of children's medicine |
| 2013 | NewCanuck | Helping immigrants learn local culture |
| 2014 | Speech Coach | Helping speakers give better talks |
| 2014 | Surgical Trainer | Measuring surgeon's movements |
| 2014 | Critter | Social Connections with Virtual Pets |
| 2014 | MyAlly | Helping stressed/suicidal teenager's |
| 2015 | flapCheck | Monitoring recovery from plastic surgery |
| 2015 | Peptiblocks | Game to find good protein folding |
| 2015 | PUPL | Pupil light reaction measurement |

Table 2: Selected Projects

## 7. Issues and Future Offerings

We have been very pleased and excited by the various outcomes of the course, but there are a number of issues with this kind of course that we list below, together with some thoughts as to how we will revise the course to deal with them.

One key isssue relates to the Appers in the course – they are required to do a fair bit of work at the beginning in terms of the assignments described in Section 5, and the ideation and proposal phases of the project. However, after that, the amount of work and contribution by the Apper varies widely. In the best cases, the Appers engage fully, doing the UI design, active testing and feedback of various versions of the software, comparison with laboratory measurements, and sometimes even some programming. In other cases, there is less active involvement and the Programmers have largely taken over even some of the specification work. It is difficult to prevent the latter, but it can be mitigated by careful vetting of the Appers early on in the course, to ensure that they have sufficient expertise and capability to contribute; we have noticed a correlation between the expertise level of the Appers and their ability to contribute.

A second issue is the quality of the feedback that we provide the students. While there is quite a bit of feedback given verbally after each presentation when urgently needed, and some written feedback, we feel that students could benefit from more private and individual feedback and guidance. This arises because of the very un-directed nature and broad nature of the work in the project – for some students, this is one of their very few experiences of this kind (rather real-world) work. In a sense, each project is like a miniature master's thesis, and there is room for far more constructive feedback. With the scale of students in the course it is difficult to provide this level of feedback. We are planning to re-think methods of providing better feedbak.

A third issue is recruiting of students. The University of Toronto is one of the largest campuses in North America, with 65,000 students total and 16,000 graduate students, in many different departments. In the past we have 'marketed' the course by speaking with Associate Chairs of Graduate Studies in many departments and faculties, and conveying email advertisements. We have begun to feel that there are many more highly qualified expert Appers who would be interested in the course, who have not heard about it. Our plan going forward is to offer seminars in different departments to describe the course and outcomes related to that field.

## 8. Conclusions

We have described a novel graduate course that brings together students from many disciplines to prototype mobile and wearable applications in those different fields - working collaboratively with graduate-level Programmers. The outcome has been a very exciting experience for both the students, the instructor and the teaching assistants, with many novel ideas explored. We have also found that a great deal of integrative learning takes place, along with inter-discplinary thinking. There is also good experience in communication across disciplines. This paper (and associated web sites) describe the nature and structure of the course, which could easily be taught in any graduate-level University with programming and other disciplines. Our hope is that this document could enable it to happen many times over!

## 9. Acknowledgements

The authors would like to thank the many students who took this course and worked hard to get a great result. We are also grateful to Google and Huawei for donations of smartphones, and Arshia Tabrizi, esq., for a donation to support the course.

## References

[1] P. Hinds, K. Carley, D. Krackhardt and D. Wholey, "Choosing Work Group Members: Balancing Similarity, Competence, and Familiarity," Journal of Organizational Behavior and Human Decision Processes", Vol. 81, No. 2, pp. 226–251, 2000.
[2] B. Nejmeh and D. Weaver, "Leveraging scrum principles in collaborative, inter-disciplinary service-learning project courses," IEEE Frontiers in Education Conference (FIE), pp.1-6, Oct. 2014.
[3] (2013) ECE 1778 Final Presentations 2013. [Online]. Available: http://uoft.me/1778-2013
[4] (2014) ECE 1778 Final Presentations 2014. [Online]. Available: http://uoft.me/1778-2014
[5] (2014) ECE 1778 Final Presentations 2015. [Online]. Available: http://uoft.me/1778-2015
[6] (2015) ECE 1778 Course Website 2015. [Online]. Available: http://uoft.me/1778-2015front
[7] M. Poh, D. McDuff, and R. Picard, "Non-contact, automated cardiac pulse measurements using video imaging and blind source separation," Opt. Express 18, 10762-10774 (2010).
[8] TI Sensor Tag [Online]. Available: http://www.ti.com/sensortag
[9] Variable Node Device [Online]. Available: http://variableinc.com/products/
[10] TrackR Device [Online]. Available: https://www.thetrackr.com
[11] M. Murphy, "The Busy Coder's Guide to Android Development," [Online]. Available: https://commonsware.com/Android/
[12] J. Nutting, F. Olsson, D. Mark and J. LaMarche, "Beginning iOS 7 Development, Exploring the iOS SDK," Apress 2014.
[13] MIT Scratch [Online]. Available: https://scratch.mit.edu
[14] Go Mockingbird UI design tool [Online]. Available: https://gomockingbird.com
[15] Moqups UI design tool [Online]. Available: https://moqups.com
[16] Pencil UI design tool [Online]. Available: http://pencil.evolus.vn/Stencils-Templates.html
[17] Designing for iOS [Online]. Available: http://tinyurl.com/pxxh66k
[18] App Structure [Online]. Available: http://developer.android.com/design/patterns/app-structure.html
[19] N. Shah, L. Carvalho, and I. So, iAnkle video and report [Online]. Available: http://uoft.me/iAnkle
[20] Z. Teitel, V. Chen, F. Zhao, Baton video and report [Online]. Available: http://uoft.me/Baton
[21] E. Guy, S. Puri, and S. Chen, Mindful Me video and report [Online]. Available: http://uoft.me/MindfulMe

# Agile Systems Integration Process

**Jason D. Winningham, David J. Coe, and Jeffrey H. Kulick**
Department of Electrical and Computer Engineering
University of Alabama in Huntsville, Huntsville, Alabama, USA

**Abstract -** *This paper presents an adaptation of an Agile development process specifically tailored towards software systems integration projects. The proposed Agile Systems Integration Process has been piloted in a senior-level software engineering studio course using an HO gauge model of a Positive Train Control system as the cyber-physical target hardware. The software developed by the students included five components: train scheduling, interactive display control, walk-about tablet manual controller, track geometry and connectivity input and safety monitoring software. Approximately 30 students were divided into ten teams with two teams competing to build each of the five components. This paper discusses what aspects of the proposed Agile systems integration process worked successfully and several important lessons learned to improve future performance.*

**Keywords:** Agile, systems integration, positive train control, agile in an educational environment

## 1 Introduction

The classical approach to software development is called the Waterfall process, which is shown in Figure 1 below [1]. This paradigm is often used in industries that have large complex systems that have to be sustained over many years such as aircraft, weapons systems, and cyber-physical systems. Oftentimes the process is proscribed by a certification organization such as the Federal Aviation Administration (FAA) for aircraft systems. The currently followed processes, DO-178C calls for explicit identification of the phases and artifacts of each stage and formal reviews either by the FAA or it's representative called a DER (Designated Engineering Representative) [2]. Each stage of development has formally identified documents that must be produced as part of the states: requirements document, design document, code and unit test documents, integration documents.

At the production of each artifact, the quality of the artifact is assessed. A formal review is performed including a requirements review, design review, unit test review, integration test, and independent verification and validation review. The goal of these quality control steps is to identify defects as early as possible since the defect removal cost increases exponentially the longer the defect remains in the product [3].

The advantage of the Waterfall approach is that much thought must be given to each stage, starting with the requirements and ending in the verification and validation, that the code meets the requirements and the requirements are the correct one for the system. However, as we all know, system requirements change as a system evolves. Over the multi-year development cycle of a weapon system, for example, technology evolves and threats emerge that were unforeseen at the time the program started.

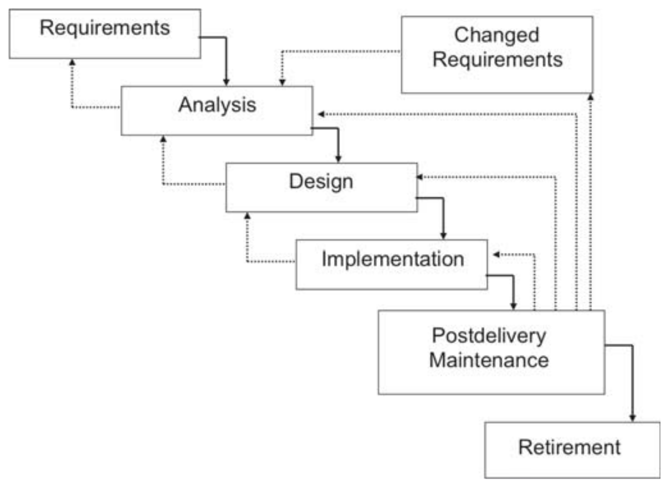



Figure 1 – Classic Waterfall Process [1]

A modified Waterfall model, called the spiral model has evolved to allow for a cyclic evolution of complex products [4]. As shown in Figure 2, each phase is run to completion with a delivered product following which the requirements are re-evaluated, the design re-considered and new requirements and design elements added. During the multiple cycles of the spiral process full-fledged artifacts are produced including requirements, design, code and test, and integration test documents.

While these processes are appropriate and even required by many industries, the new more responsive Agile approach has been developed for applications that do not have the need for detailed artifacts or reviews. Agile development processes are often used by small teams of developers for short lifetime or short turnaround products such as cell phone applications or internal research and development projects. In an Agile process such as Scrum, requirements are replaced by user stories and the development cycle is very short, called sprints. Often these sprints are just a few weeks long. At each sprint the development team consults with the customer, identifies part of the product that can be produced in the next sprint, and develops and tests the code for that sprint. A key element of Agile processes is the delivery of an implemented and tested partial product at the end of each sprint.

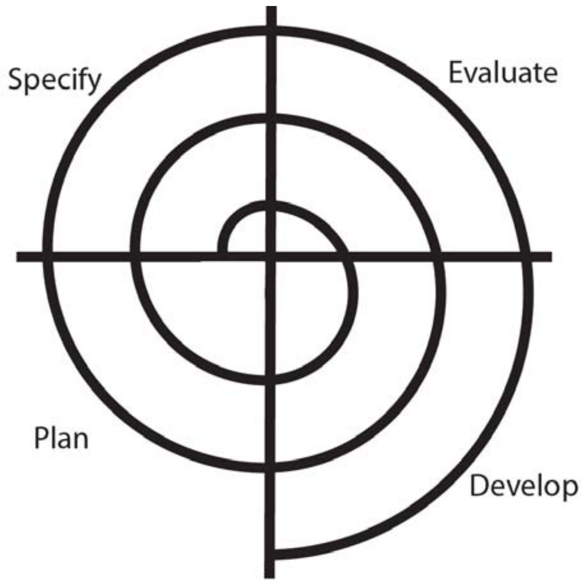



Figure 2 – Spiral development process [4]

One huge advantage of the Agile approach is that defective or misunderstood requirements are very quickly identified since the intended interpretation of the requirements are revealed within a short period. There is no question that defective or conflicting requirements are rapidly identified and remediated.

As development of large cyber-physical systems have evolved, which have traditionally used a Waterfall process, developers have been using modeling, simulation, and hardware in the loop for systems of systems integration to deal with the misalignment of development times and delivery dates for completed subsystems. So it is not uncommon for a complex system to be built completely virtually, instantiated in a hardware-in-the-loop framework (HWIL), and have component parts replaced piece by piece (when available). Even after the component parts are available, virtual implementations may continue to be used due to limited availability of hardware or software components. Thus complex missile systems may have started with some legacy hardware, simulation models of threats, sensor behavior, command and control components, and synthetic threat model generators. As the system progresses, components of the HWIL system will be replaced with real hardware/software as part of the system integration activities. Parts, such as threat simulations may remain in their initial form as Agile developed components so that they may be rapidly adjusted to model evolving threats and sensor capabilities. So there is a need for systems of systems integration with both fully Agile developed components and mixed systems with traditional waterfall and Agile developed subsystems.

In this framework the University of Alabama in Huntsville evolved an existing agile development senior software design studio into a systems of systems integration experience using Agile-developed parts by multiple teams of developers. Below we first present a description of the senior software studio course, the cyber-physical system utilized during the pilot study, and the Agile systems integration process as employed in the pilot study.

## 2 Agile senior software studio course

Our senior-level software engineering studio course was chosen to pilot the Agile systems integration process. This one semester course was originally created to provide an immersive team software development experience for undergraduate computer engineering students at the University of Alabama in Huntsville (UAH). In this course, teams of three to four students design, implement, and test a non-trivial software product. As the course was initially formulated, students employed a Waterfall methodology that included the preparation of several milestone documents including the Software Requirements Specification (SRS), Software Design Document (SDD), and Software Test Plan (STP) in addition to development and testing of the software itself. After several offerings of the course, it became clear that one semester was too little time for some students to develop a high quality non-trivial product using a Waterfall methodology.

In early 2011, a more Agile process derived from Scrum was adopted for use in the course. In developing this process, it was important to retain some elements of a Waterfall process such as traceability and test coverage analysis since many graduates from UAH remain in north Alabama developing safety-critical systems such as helicopters, missiles, and radar systems. It was also important to reduce documentation demands so that students had more time to implement and test their products.

At the beginning of the course, students are divided into teams of three to four students. Every student on the team develops his or her own complete proposal for a team-sized project that satisfies the list of project constraints, which include development of an open source product, use of real-time interactive graphics, the use of socket-based networking, interaction with an SQL database, and mandatory use of Subversion repository. Given the constraints, students typically pursue development of a video game or simulation. The project proposal includes the following sections.

- Project Overview
- Market Research
- Fundamental Requirements
- Storyboard
- Derived Requirements
  - High-level use cases
  - Non-functional requirements
- Traceability

After receiving feedback from the instructor on the proposals, each team must select the project they want to pursue from the set of the project proposals produced by members of that team.

The modified Scrum process includes the following five steps, which are further explained below.

- Sprint planning
- Sprint execution
- End of sprint live demonstration
- Sprint retrospective
- Peer evaluation

During the semester, students have time to execute four to five sprints.

In the *sprint plan*, students must include detailed use cases only for those use cases selected for the sprint backlog. Use cases not selected for the sprint backlog remain on the product backlog as high-level (unelaborated) use cases. Additionally, students must include a detailed set of acceptance tests for all use cases on the sprint backlog (test-driven development). Note that by requiring detailed use cases instead of user stories, the use cases may serve as both requirements and as test scripts for conducting the end of sprint acceptance tests.

At the end of each sprint, students must conduct a *live demonstration* of the new functionality implemented. For the live demo, the team must check out their source code from the Subversion repository, compile it, and execute the set of acceptance tests along with any additional tests suggested by the instructor or customer. Responsibility for conducting the demonstration rotates through the team members so that everyone must demonstrate competence and participation in the project.

The traditional *sprint retrospective* provides an evaluation of the process as executed during that sprint. The retrospective for the modified Scrum process has been augmented with additional information including a labor report itemizing the time spent by each team member and productivity statistics such as source lines of code produced and use cases attempted versus completed. It also includes Subversion repository statistics gathered using the StatSVN utility to indicate who is or is not contributing to the project [5]. Finally, the retrospective includes a summary of the acceptance test results (pass/fail).

At the end of each sprint, each student completes a *peer evaluation* form (derived from [6]) in which they must assess the performance of each team member on four criteria: Quality of Work, Quantity of Work, Level of Teamwork, and Timeliness of Work. The performance scale ranges from 1 (little or no contribution) to 5 (exceptional). Written comments are required to explain any below average assessment of 1 or 2. After completion of the source code development sprints, a final "documentation" sprint is performed during which document deliverables are updated and completed to match the as-built product.

While the implemented and tested source code within the Subversion repository was the primary course deliverable, students were also required to deliver a set of requirements, which consisted of the set of as-implemented use cases (updated from those in the sprint plan), as well as a test report containing the acceptance tests developed for each sprint (from the sprint plan) and the test results (from the sprint retrospective). Finally, as a measure of the thoroughness of their testing efforts, students are also required to deliver a test coverage report that summarizes the amount of delivered code per module that remains untested.

# 3 The cyber-physical system

For the pilot systems of systems integration study, students were asked to develop train scheduling, display control, walk-about tablet manual controller, and safety-monitoring software for our newly enhanced Positive Train Control (PTC) test bed.

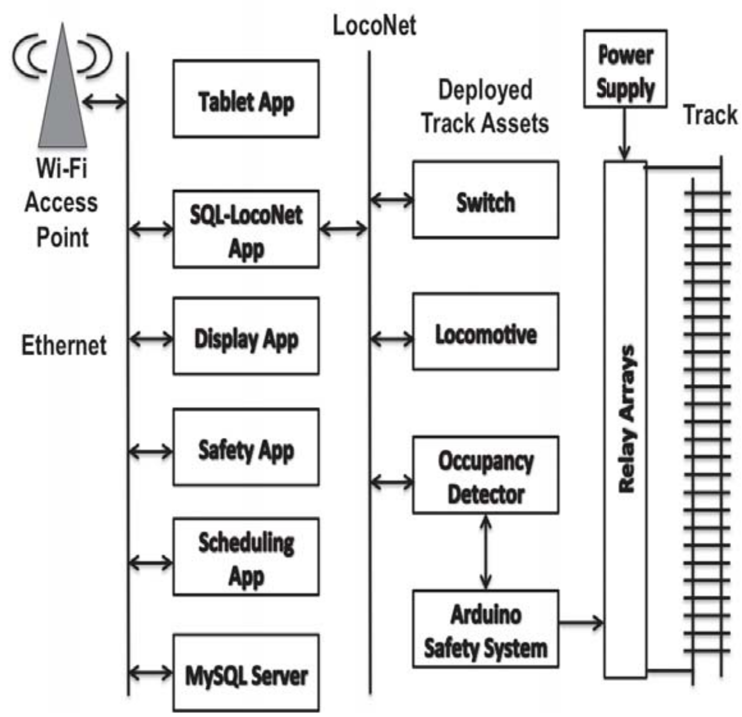



Figure 2 – Block diagram of system

The components of the system as identified in Figure 2 include:

1- a train scheduling system that computes routes for trains to follow and issues commands to trains and switches as the train progresses down the route;

2- a safety monitoring system that implements what is tradiditionally called vital logic and is concerned with local safety issues such as trains heading towards a collision;

3- a display system that displays for hte user the state of the tracks, train occupancy, train routes, switch settings, and safety violations.

4- a track layout system that allows the entry into the train control system layouts of tracks, switches, and detection sections, and

5- a tablet based controller for manual remote control of the layout's locomitives and switches.

As previously presented [7-10], the UAH Software Safety and Security Engineering Laboratory has developed HO gauge model train cyber-physical systems of increasing complexity to facilitate our teaching and research programs in software safety and security. Our PTC test bed models real world positive train control systems currently being deployed nationwide in the United States. A key feature of these real world PTC systems is the centralized safety monitoring system that can remotely control trains in order to prevent accidents.

The five primary software components of the system (routing, safety, display, layout and manual trablet control) are interfaced to each other and the track via a SQL server wiich maintains a set of SQL tables that reflect the track status (as read from the track) and track commands (as issued by the various control components)

The trains and track switches are controlled by and observered by a set of Digtrax LocoNet hardware which silultantously monitors locomotive location while issuing commands to the locomotives (speed, direction) and switches (through, bypass). The component that reaches across the SQL database and LocoNet hardware is called the SQL-LocoNet bridge.

As shown in Figure 4 below, our PTC test bed has been recently re-engineered to increase the total number of individually-controllable blocks of track to over 100 blocks. The track layout consists of two different train systems, a main line track consisting of the outside ovals and rail yards and the interurban system that consists of the figure-eight loops connected by track that crosses the main line track at four crossover points. The added complexity of these crossovers present points of interference that must be accounted for in both the scheduling system and the safety-monitoring systems.

An Arduino-based block occupancy detection and safety system has recently been integrated into the PTC test bed. Redundant occupancy detection is employed to prevent collisions in case of hardware failures. As with real-world trains, current sensing is employed to detect the presence of a locomotive within a specific block of track [11]. As previously reported, fine grain localization will be provided wirelessly by inertial measurement unit/tie counting transponders located on each train [9]. The re-integration of fine grain localization into the enhanced PTC test bed has been left to the next offering of the software studio course.

Figure 4 – Panoramic photograph of PTC test bed [9].

As a safety precaution, the operating rules of real train systems require that only one train may occupy a given block of track, except under certain special circumstances. Figure 5 below shows one of the two Arduino boards that comprise the safety system. Each block of track in the PTC test bed has its own power supply that has been routed through a relay mechanism to allow the safety system to shut off that block’s power supply to prevent an impending collision. Figure 6 below shows the custom-designed wiring consolidation circuit board (purple) that integrates multiple off-the-shelf Digitrax controller boards (green) with the arrays of relays used by the safety system to shut off power to blocks of track. Thus, even with a defect in the scheduling software failure, the trains should never collide. Short circuit protection is offered by fuses located in series with each relay.

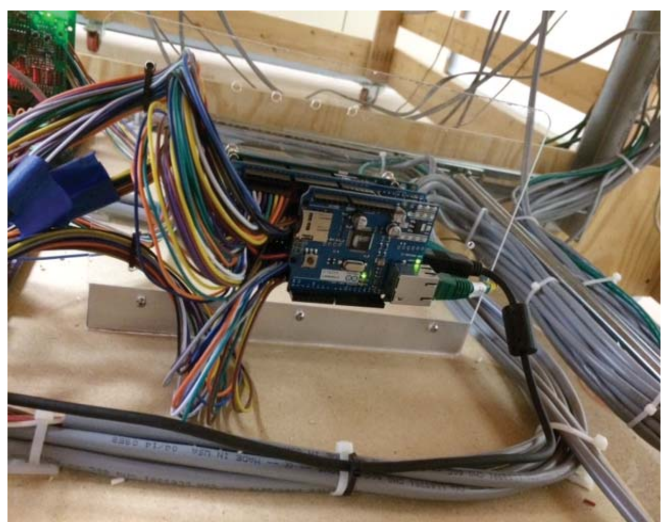


Figure 5 – Photo of Arduino-based safety monitoring system

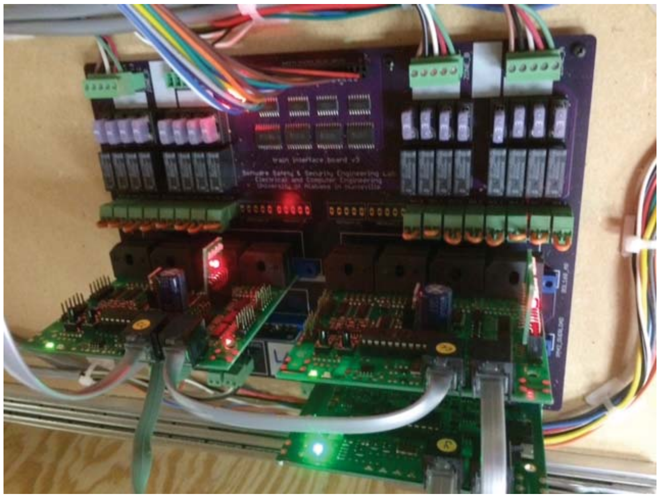


Figure 6 – Photo of safety system relay arrays.

# 4 Pilot Systems of Systems Integration Study

## 4.1 Setup

As discussed earlier, the software development task was divided up into five parts with an SQL database acting as the interface mechanism. By making the interface between the components a database, it was possible for each team to develop and test their code as it evolved to ensure it met the interface description of the database tables. The large enrollment in the course (30+ students) resulted in a total of ten development teams of 3-4 students each. To increase the likelihood that a fully functional, five part system would be developed, two teams, an A team and a B team, were assigned to develop competing implementations for each of the five parts.

A key requirement for the fully integrated system was that all parts delivered must be interoperable with each other so that all teams would have to interact with each other to develop appropriate component interfaces. For example, the train scheduler developed by the A team must be interchangeable with the train scheduler developed by the B team in the fully-integrated system. Moreover, any part 1 component must be compatible with all parts 2 – 5.

## 4.2 Systems Integration Plan

The intention at the onset of the project was to use continuous integration which is a primary property of both agile integration and traditional integration processes. In continuous integration, components and builds of components are integrated as they become available. Sketches of a component's capabilities are to be serially replaced by their ever refined version as time goes on. If some component lags the others in developmental maturity, legacy components could continue to be used. The goal was to have each of the teams performa an integration activity with the code that was developed after each sprint. Because the integration was managed by the SQL-LocoNet bridge, an asynchronoously updated database, students could mix and match old and new (tried true or developmental) components as they wished.

# 5 Observations and Discussion

Although the systems integration plan seemed reasonable at the time, a number of issues arose that led to a somewhat different execution.

Agile processes call for replacement of formal documentation by personal communication in the form of daily meetings and bi-weekly retrospective/prospective planning meetings. The students in the class are somewhat different than your typical undergraduates in that most of them have full time jobs in the local industrial and research community that is supported by the US Army's Redstone Arsenal and NASA's Marshall Space Flight Center. Typically the only one-on-one physical meetings occurred after class which ended at 7pm two times a week.

Generally the meetings that did occur were too short and too dense (30 students attended class regularly of the 32 enrolled). As a result the absence of formal documentation and the absence of regular inter and intra team meetings resulted in numerous teams moving forward with development before fully developed ICDs describing the interface were created. Conomintantly, students developed unit test methodologies for their personal incarnation of the data base. Although several class sessions were dedicated to the emerging bifurcation of the database standards, differences remained until the very end of class when integration testing was performed as a formal requirement.

Another issue that arose was the continuous evolution of the target hardware. An early version of the hardware was available at the beginning of the term for student use. However, the final complete layout with all its whistles and bells was not available until the second half of the term. The students failed to appreciate the fact that such issues arise frequently in the development of real world systems.

During the study, the university was also closed unexpectedly due to several snow days. The university was closed for up to two weeks equivalent workdays which was unprecedented in the previous 25 years history of the authors. Students used these periods of reduced communication to forge ahead on their own vision of the interfaces, which resulted in more aggravation and breakage of code.

Finally, some methodologies used by students, such as copy and paste of large code segments rather than writing a parameterized function or developing table driven code, resulted in at least one team producing 25,000 source lines of code. While the code was highly repetitive, each defect detected required the repetitive repair of each block of copied code for each defect detected, which likely increased the probability that one of the students might have missed applying the repair in one or more places.

## 6 Conclusions

The Agile course upon which this paper is based is undergoing evolution as we speak and the success of the endeavor has only been demonstrated a few days ago. Of the 10 team development projects, 9 were completed all the way through systems integration. The goal to have students develop a large (each component ran thousands of source lines of code) multicomponent project which involved component development and system integration based on agile processes was much harder than expected. Even though the course is ongoing, several important lessons have been learned for future enhancements to the course.

*Lesson Learned*: *Face-to-face communication must occur*

Students' rapid and eager acceptance of low-documentation processes was not coupled with the need for much greater verbal and meeting mediated communication. Agile replaces detailed specification and design documentation with face-to-face communication. The integration plan hinged on these face-to-face communications to resolve component interface issues early on. Since some teams failed to conduct these face-to-face discussions and other teams failed to abide by the decisions made during these discussions, interface issues continued to arise even in the last sprints. For interfaces between teams the old fashioned signed off ICDs would probably have been a better implementation. For processes that require significant communication within and between team members communication vehicles such as Skype and Google drive should be set up at the very earliest stage. Moreover, points must be assigned to these communications as incentive for students to engage in them.

*Lesson Learned*: *All code is owned by the team*

Students who have not worked in an environment where breakage is a measure of the success of the project rather than the loss of an individual's personal code contribution to the project. Agile is predicated on far more rework and scrapping of codes that are developed earlier in the process. Students feel that each line of source code is fully formed work product are loathe to scrap or rework them. Agile processes gain their ability to identify and rectify early errors only through much larger breakage in the work product than processes with great planning and breakage should be looked upon as progress rather than lost children.

*Lesson Learned*: *Practice continuous integration within and across components.*

Continuous integration is required for success of agile based systems of system integration. Not only must each team develop working, testable product each sprint, these must be integrated on a continuous basis to enforce acceptance of interface specifications rather than a wild west approach to interface design (my way or else).

## 7 Acknowledgements

We would like to thank Dr. George Petznick, our resident train expert, for his assistance in the design and construction of the Positive Train Control test bed.

## 8 References

[1] Stephen R. Schach, *Object-Oriented & Classical Software Engineering*, 7th edition, McGraw-Hill, 2007

[2] RTCA DO-178C, "Software Considerations in Airborne Systems and Equipment Certification", December 13, 2011.

[3] S.H. Kan, S.D. Dull, D.N. Amundson, R.J. Lindner, and R.J. Hedger, "AS/400 Software Quality Management," *IBM Systems Journal*, vol 33, no. 1, 1994.

[4] B.W. Boehm, "A Spiral Model of Software Development and Enhancement," *IEEE Computer*, vol. 21, May 1988.

[5] StatSVN utility. http://www.statsvn.org/

[6] David Kelly and Mary Sadowski, "Peer Evaluation within a Team Design Project", ASEE Conference Proceedings

[7] David J. Coe, Joshua S. Hogue, and Jeffrey H. Kulick, "Software Safety Engineering Education," 2011 International Conference on Software Engineering Research and Practice (SERP'11), WORLDCOMP 2011, July 18-21, 2011, Las Vegas, NV.

[8] Travis Cleveland, David J. Coe, and Jeffrey H. Kulick, "Video Processing for Motion Tracking of Safety Critical Systems," 2013 International Conference on Software Engineering Research and Practice (SERP'13), WORLDCOMP 2013, July 22-25, 2013, Las Vegas, NV.

[9] Scott Schiavone, Sjohn Chambers, Sunny Patel, Lee Ann Hanback, David J. Coe, Jason Winningham, B. Earl Wells, George Petznick, and Jeffrey H. Kulick, "UAH OnTrack: Precision Navigation System for Research on The Software Safety Issues of Positive Train Control," 2014 International Conference on Software Engineering Research and Practice (SERP'14), WORLDCOMP 2014, July 21-24, 2014, Las Vegas, NV.

[10] David J. Coe and Jeffrey H. Kulick, "Positive Train Control: Concepts, Implementations, and Challenges," 2014 International Conference on Software Engineering Research and Practice (SERP'14), WORLDCOMP 2014, July 21-24, 2014, Las Vegas, NV.

[11] American Railway Association, *The Invention of the Track Circuit*, New York, 1922.

# Smart Chair: An Internet of Things Case Study for a Capstone Research Project

**Jing (Selena) He, Amir Atabekov, Edward Mwangi, and Donald Toler**
Department of Computer Science, Kennesaw State University, Kennesaw, GA, USA
jhe4@kennesaw.edu; {aatabeko, emwangi, dtoler1}@students.kennesaw.edu

**Abstract**— *This paper describes the experience of a team of two undergraduate students and one graduate student to design and implement an Internet of Things (IoT)-based research project in a final year capstone project class at Kennesaw State University. The Internet of Things involves connecting physical objects to the Internet, which provide opportunity to build intelligent systems and applications by leveraging Radio-Frequency Identification (RFID), Near Field Communication (NFC), Wireless Sensor Network (WSN), and universal mobile accessibility advanced technologies. In this case study, the office/lab chairs are proposed to connect to the internet. During the one semester's capstone project class, the proof-of-concept implementation (*i.e.*, connecting one chair to the internet) is completed. To be specific, an assembled Ardunio system is responsible for scanning user ID and obtaining chair occupancy status and then sending that information to the cloud server. The collected and stored data on the cloud can be retrieved and displayed on an Android application.*

*Nowadays, the undergraduate degree program has gradually evolved to include research projects in the capstone project class. Moreover, there is a growing need in our local industry for graduates with research skills in advanced technologies. This Internet of Things case study shows a good example of learning advanced technologies during the research process so that students can keep pace with the rapid rate of change in computer science.*

**Keywords:** Internet of Things, Capstone Projects, Agile Software Development, Ardunio Boards, Cloud Services, Mobile Applications.

## 1. Introduction

Capstone projects are culminating experiences situated in the final year of an undergraduate students college curriculum and are designed around students demonstrating mastery of both content and application of relevant subject matter. In computer science, project-based learning model is often adopted in Capstone projects, which focus on developing case-specific problem understanding to create feasible solution options [1], [2]. In a project-based learning model, professional project-managements approach provides steps and tools to structure and support students' work. Nowadays, a growing proportion of such capstone projects undertaken at Kennesaw State University (KSU) have had a research focus, since KSU changed its status to *comprehensive university* in August 2013. Except for facilitators, the role of instructors extends to include proactive guidance of students in the challenging process of constructing not only new understanding but also developing feasible solutions in capstone research projects. This case study's smart chair system describes one such research project which connects office/lab chairs to the internet. The data of who occupancy the chair, and occupancy time duration will be collected and stored in the cloud server. The stored data can be accessed by authorized users at anytime anywhere. Moreover, the analyzed data can be used in various commercial/educational systems such as resource management, tutor time tracking management, students attendance checking, dynamic ticketing, and so on. Because of the time limitation, agile software development methodology [3] are adopted in this capstone research project. Agile methodology is described as iterative and incremental "inspect-and-adapt" approach, which provides opportunities to assess the direction throughout the development lifecycle. Every aspect of development, such as requirements, specification, design, etc. is continually revisited, so that agile methodology greatly reduces development costs and time [4].

The effective management of classrooms, halls, offices, and public spaces in any organization and the efficient and effective deployment of an organization's resources when they are needed are challenging problems. How to automatically record the activities undertaken and monitor resources usage inside rooms in real-time is usually intricate. The smart chair system described in the paper tries to solve the problem by applying Internet of Things techniques [5], [6], [7], [8], [9], [10], [11] to create an intelligent environments or decision making environments [12], [13] to facilitate the use and management aspects of our daily life. Taking students' attendance monitoring systems as a motivational example, in the past, instructors had to call the names of students or the students had to sign their names on the attendance check sheets. The former is a time consuming process, while the latter is unreliable and the attendance sheets could be lost or damaged. Subsequently, facial and voice recognition methods are proposed to verify the identity of students. Recently, Radio-Frequency (RF) communication based methods [14], [15] and smartphone based methods [16], [17] are extensively investigated. The former has some limitations to cover the whole area of a classroom by applying Near Field Communication (NFC) or Bluetooth technologies; while the latter requires the students click on smartphones. Different from the aforementioned techniques, the smart chair system can automatically check whether the chair is occupied or not by using remote sensing technologies. Moreover,the identification of the students can be automatically collected by using RFID technologies. Additionally, the lateness or leave-early information can be analyzed by integrating timestamps. Furthermore, the smart chair system has brilliant commercial prospects, which are helpful to build intelligent resource management systems, intelligent dynamic ticketing systems, etc.

Through successfully implementing the smart chair proof-of-concept research project, the students got familiar with the process of conducting research, which includes developing a research project; conducting a literature review; investigating appropriate methods and tools to implement the research project; making oral presentation of the application of the research project; and writing technique manuscript of the research findings. Moreover, this research project provides students with the opportunity to apply the knowledge and skills acquired in their courses to a specific practical problem; extend their academic experience into working with new ideas and learning new advanced technologies; demonstrate their proficiency in written and oral communication skills; extend and refine their knowledge and skill in the realization of their personal and professional goals.

The rest of this paper is organized as follows: In Section 2, the system architecture is specified. In Section 3, we describe the actual design and implementation of the smart chair system. Finally, the paper is concluded in Section 4 as well as the future work directions.

## 2. System Architecture

The smart chair system is based on a network of connected sensors embedded on the physical chairs to collect information, which is governed by the functionality of Internet of Things. All the collected information are uploaded to the cloud server so that any application can take advantage of data anywhere anytime when necessary. In this capstone project, we implemented the proof-of-concept experiment on a single chair. Later, we will extend it to a connected smart chair system. This project has three major components which include:

- The "thing": the chair ID and its occupancy state are stored in the cloud via Ardunio Yun [18].
- The cloud server: ThinkSpeak [19].
- The mobile application: the chair occupancy state and the user information can be displayed and monitored through mobile application.

The overall architecture of the smart chair system is described in Fig. 1. As shown in Fig. 1, RFID reader can automatically read the user's identification (denoted by UID) and passes it along to the Arduino ATmega32u4 microcontroller. The custom program checks if the pressure was applied to the pressure resistor. If the pressure applied is more than twenty pounds, the program then passes the timestamp to the Atheros AR9331 microcontroller through the bridge library [20]. Finally the bridge library issues a HTTP POST request to the ThingSpeak API [19] along with the UID of the scanned RFID card.

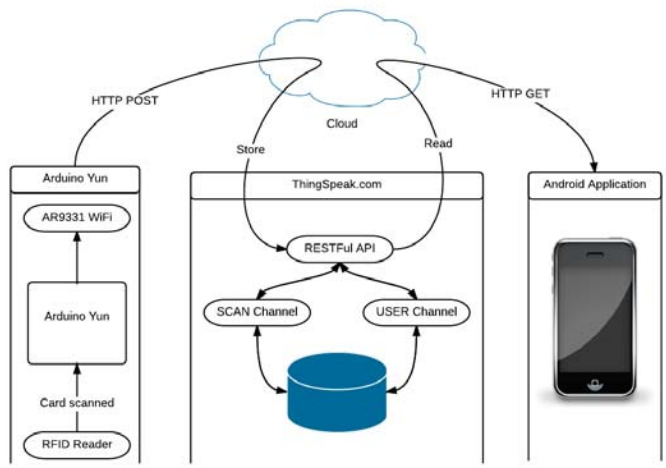



Fig. 1: Smart Chair System Architecture.

Moreover, the Android application is also developed to let administrators or other interested users monitor when or who has scanned his card and sat on the chair. The overview of the Android architecture is also illustrated in Fig. 1. The Android application issues an HTTP GET request to ThingSpeak's JSON feed. Each HTTP GET request is authenticated against an API key (more details will be mentioned in the next section). When submitting a HTTP GET request, the API key must be included in the URL string. Finally, the retrieved data will be displayed on the Android application.

In more detail, the data flow of the whole system is summarized in Fig. 2. First RFID scan is performed and the card UID is read and transferred to the Arduino microcontroller, then the program on Arduino transfers the data to the AR9331 chip with bridge library. AR9331 being a WiFi chip submits a HTTP POST request to the RESTFul API of the ThingSpeak.com. Likewise in order to display the latest scans, Android application submits HTTP GET request to the RESTFul API and fetches the data in JSON format. Finally Android application parses the JSON and displays the latest scans.

## 3. System Design and Implementation

In this section, we will mention the details to design and implement the smart chair system. Before we move to the

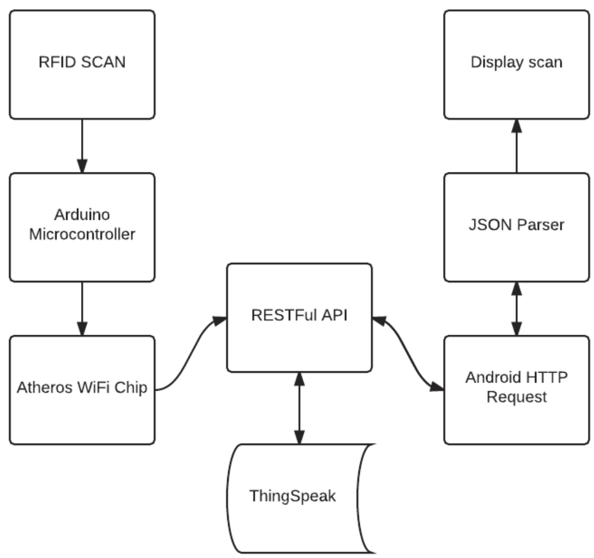



Fig. 2: System Data Flow Diagram.

technical details, the system requirements are summarized and listed in Table 1.

Table 1: Smart Chair System Requirements

| Requirement | Description |
|---|---|
| Read RFID tag UID with reader | Write Arduino program which will read the UID off the RFID tag when in close proximity |
| Sense when pressure is applied | Connect pressure resistor and write corresponding program which detects when pressure is applied |
| Program indication lights | Implement LED indication lights in the program to know the system status |
| Find and configure IoT restful service to store data | ThingSpeak chosen as backend for the project. Corresponding USER and SCAN channels will be configured to store data. |
| Send the UID data to ThingSpeak | Write Arduino program which will send the UID data to ThingSpeak channel. Yun Bridge library is used for HTTP POST requests. |
| Implement Login screen | Login screen will be implemented on Android application to authenticate users |
| JSON Parser | A JSON parser will be written to parse the JSON data returned from ThingSpeak channels. |
| Develop Android Client | Android application will be developed to view the latest scans performed. |

Next, we are ready to illustrate the system implementation details.

## 3.1 Hardware

Fig. 4 shows all the hardware components which are assembled with Ardunio Yun (shown in Figure 3):

- RFID Reader: SainSmart RC522 RFID reader was chosen for reading UID off user's RFID cards. RC522 is easy to connect to Arduino, since a library [21] was already written to communicate with the reader through Ardunio Serial Peripheral Interface (SPI) bus [22].
- Pressure Sensor: Interlink 402 pressure resistor was chosen to detect if a person is currently sitting on a chair.
- Miscellaneous: Three LEDs (Red, Green and Blue) were used to indicate the status of data transmission or the success of card read. Three 180 $ohm$ resistors were used for LEDs and one $10k$ $ohm$ resistor was used for pressure sensor.

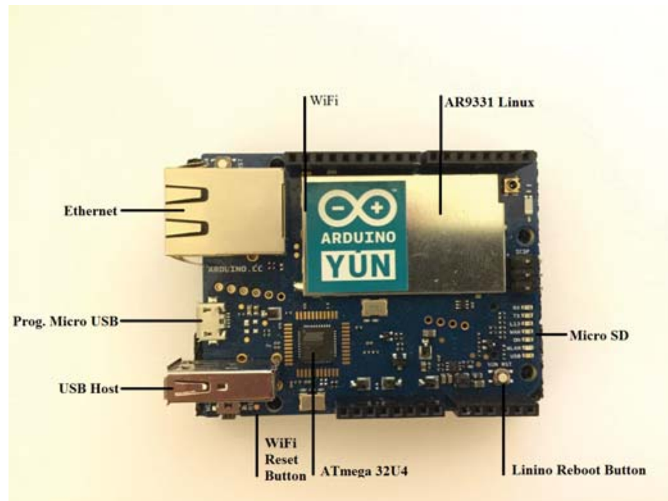



Fig. 3: Ardunio Yun Board.

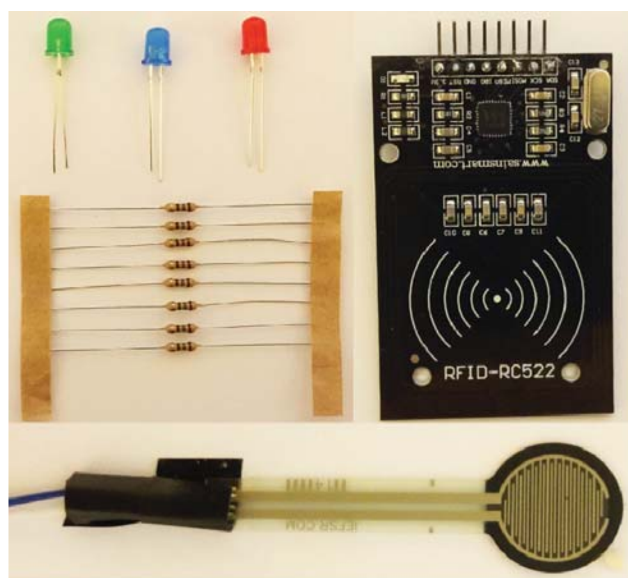
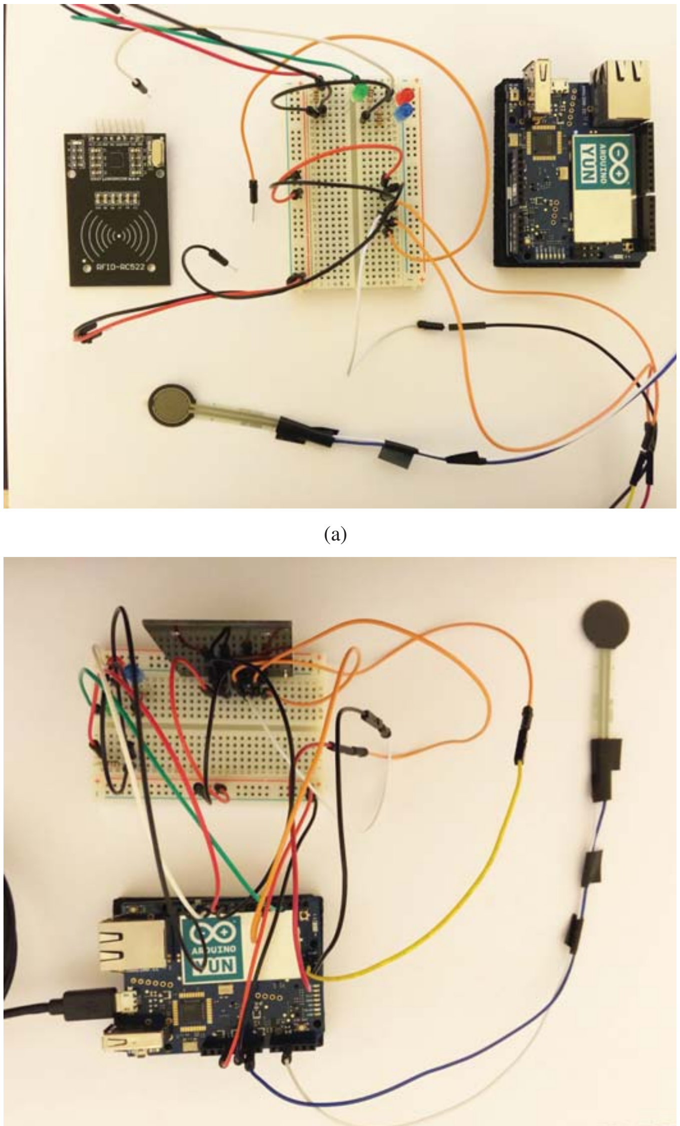



Fig. 4: Hardware Components of Smart Chair System.

The unassembled system and fully assembled system are shown in Fig. 5 (a), and (b) respectively.



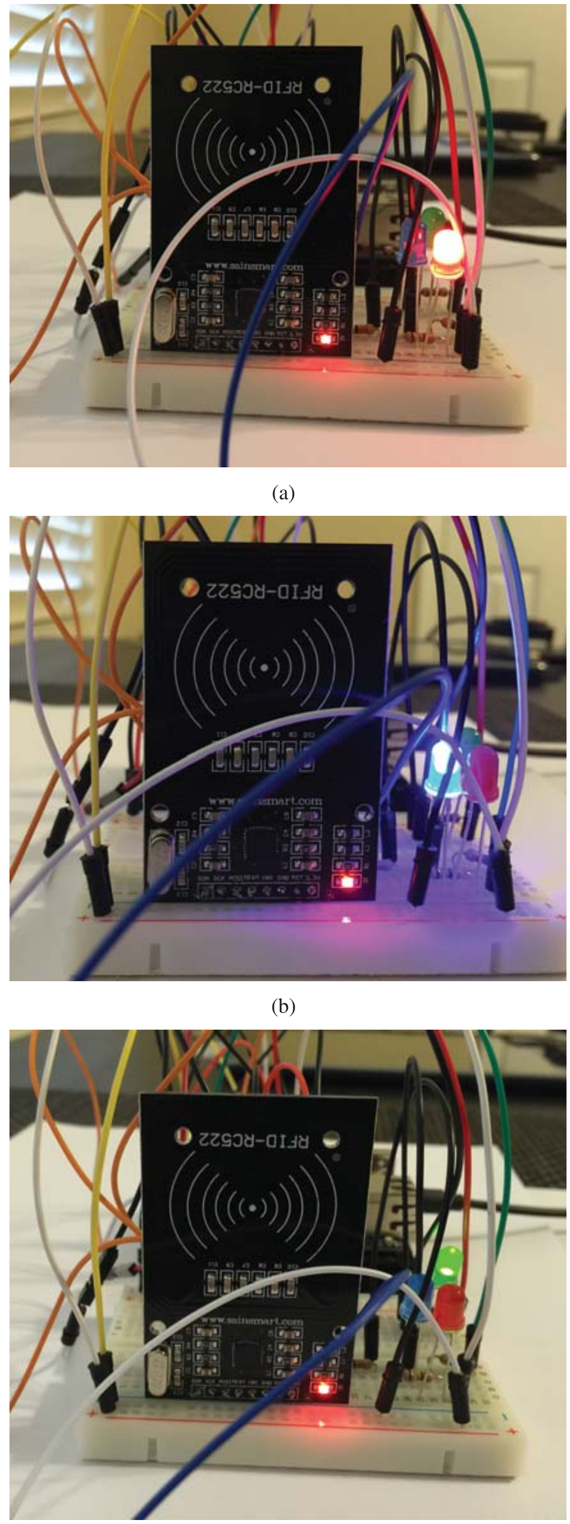
(a)

(b)

Fig. 5: (a) Unassembled System; (b) Assembled System.

## 3.2 Reading and Processing Data

The assembled smart chair system has three LEDs, which indicate success or failure of data transmission. As we already mentioned, the system will read UID through RFID reader and the pressure through pressure sensor. If the sensed pressure is greater than twenty pounds, the corresponding timestamp will be collected. As shown in Fig. 6 (a), the red colored LED indicates that the scan was unsuccessful or not enough pressure was applied to the pressure sensor. Fig. 6 (b) shows the blue colored LED is turned on, which indicates that enough pressure was applied and UID was successfully read. Finally the green colored LED indicates that data transmission to the ThingSpeak was successful as displayed in Fig. 6 (c).



(a)

(b)

(c)

Fig. 6: LED Indications in Smart Chair System: (a) RED indication; (b) BLUE indication; (c) GREEN indication.

### 3.3 Data in the cloud

Whenever there is a successful scan with enough pressure, the UID and timestamp data will be sent to ThingSpeak. ThingSpeak uses a concept of "channels", where each channel can be configured with different labels to hold different kinds of data. Our system is configured with two channels. The first channel named "USER" holds user information, while the second channel named "SCAN" holds the data related to RFID card scan. For "USER" channel we have labeled eight fields indicating card key, name, gender, email, phone number, major, address and faculty status of a user as shown in Fig. 7.

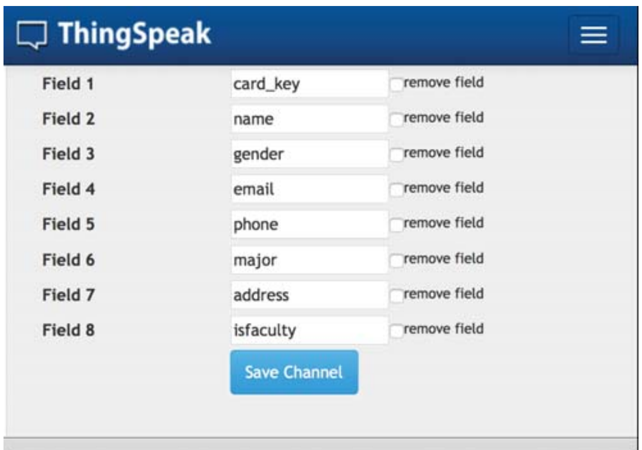



Fig. 7: USER Channel Creation Screen on ThingSpeak.

The "SCAN" channel holds only one label named "UID" along with the default label indicating timestamp. The reason is that we don't need to hold any extra information, which can easily be found in "USER" channel with the help of card UID. ThingSpeak provides a data feed for each channel in JSON format, which can be retrieved by any client application such as by Android, or iOS mobile applications. Fig. 8 illustrates the JSON feed for "USER" channel. The first object in JSON feed is called *channel*. It contains information about the channel such as the "id" of the channel, name of the channel, names of all fields in the channel and the "id" of the last entry.

Likewise the JSON field for "SCAN" channels is illustrated in Fig. 9. This channel has only one field name "UID" which holds the card UID. In addition to UID, time stamp field is provided by default. There's no need to store any more information other than UID, since the data related to particular user can be located in "USER" channel.

The first object in "SCAN" channel as always, contains information about the fields of the channel, its name and id. The second object which is a JSON array named "feeds" contains the actual feeds of the channel.


```
{
    "channel": {
        "id": 27566,
        "name": "USER",
        "field1": "card_key",
        "field2": "name",
        "field3": "gender",
        "field4": "email",
        "field5": "phone",
        "field6": "major",
        "field7": "address",
        "field8": "isfaculty",
        "created_at": "2015-02-23T01:31:52Z",
        "updated_at": "2015-03-17T14:12:14Z",
        "last_entry_id": 2
    },
    "feeds": [
        {
            "created_at": "2015-03-17T14:11:15Z",
            "entry_id": 1,
            "field1": "E3 CC 50 CE",
            "field2": "Amir Atabekov",
            "field3": "Male",
            "field4": "sample@gmail.com",
            "field5": "222-777-2187",
            "field6": "Computer Science",
            "field7": "1501 Miramar St, Valdosta, GA, 31602",
            "field8": "No"
        },
        {
            "created_at": "2015-03-17T14:12:14Z",
            "entry_id": 2,
            "field1": "94 CB 59 CB",
            "field2": "Selena He",
            "field3": "Female",
            "field4": "sample@gmail.com",
            "field5": "222-777-0000",
            "field6": "Computer Science",
            "field7": "100 Chastain Rd, Kennesaw, GA, 30102",
            "field8": "Yes"
        }
    ]
}
```


Fig. 8: JSON Feed for USER Channel.


```
{
    "channel": {
        "id": 27574,
        "name": "SCAN",
        "field1": "UID",
        "created_at": "2015-02-23T04:24:53Z",
        "updated_at": "2015-03-10T20:02:48Z",
        "last_entry_id": 215
    },
    "feeds": [
        {
            "created_at": "2015-02-24T02:07:02Z",
            "entry_id": 116,
            "field1": "E3 CC 50 CE"
        },
        {
            "created_at": "2015-02-24T02:07:17Z",
            "entry_id": 117,
            "field1": "E3 CC 50 CE"
        },
        {
            "created_at": "2015-02-24T02:07:32Z",
            "entry_id": 118,
            "field1": "E3 CC 50 CE"
        },
        {
            "created_at": "2015-02-24T02:24:16Z",
            "entry_id": 119,
            "field1": "E3 CC 50 CE"
        },
        {
            "created_at": "2015-02-24T04:09:59Z",
            "entry_id": 120,
            "field1": "E3 CC 50 CE"
        },
        {
            "created_at": "2015-02-24T04:15:39Z",
            "entry_id": 121,
            "field1": "E3 CC 50 CE"
        },
```


Fig. 9: JSON Feed for SCAN Channel.

### 3.4 Retrieving the data

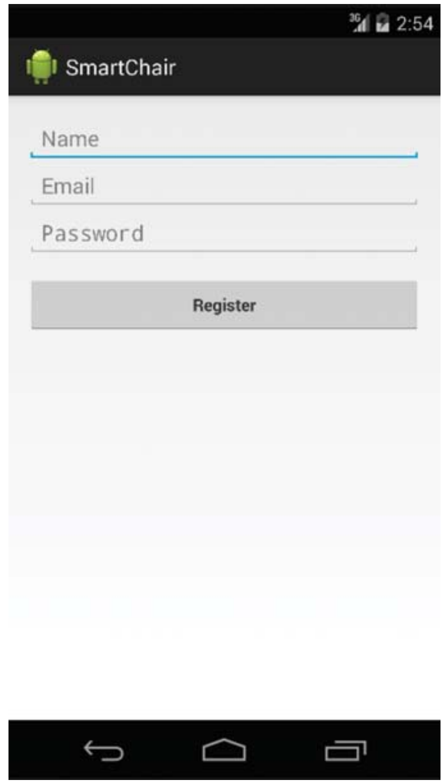
In this project, we also implement Android Application to retrieve data stored in the cloud server. As we mentioned before, the Android application needs to issue HTTP GET requests to ThingSpeak's JSON feed. Each HTTP GET request is authenticated against an API key which is generated when a new channel is configured (shown in Fig. 8 and 9). When submitting a HTTP GET request, the API key must be included in the URL string. Since ThingSpeak requires API keys to be included in every request, it is helpful to implement some form of authentication on Android application as well. Fig. 10 (a), and (b) show the registration and login screens of the Android application respectively.

Once the "Sign in" button is pressed, an Android AsyncTask is executed in the background to first fetch the results of the "USER" channel. Once the "USER" channel feed is obtained, it is stored in a HashMap data structure locally. Then a second HTTP GET request is sent to fetch the results of the "SCAN" channel. Once obtained, the data from the "SCAN" channel is merged with data we stored earlier for "USER" channel. Finally the merged records are displaced as illustrated in Fig. 11. As shown, the name of the user is displayed followed by the timestamp when the last card scan was performed. The yellow progress bar indicates the loading progress of retrieving the data. If it is full, it means that all the data was successfully fetched and stored locally.

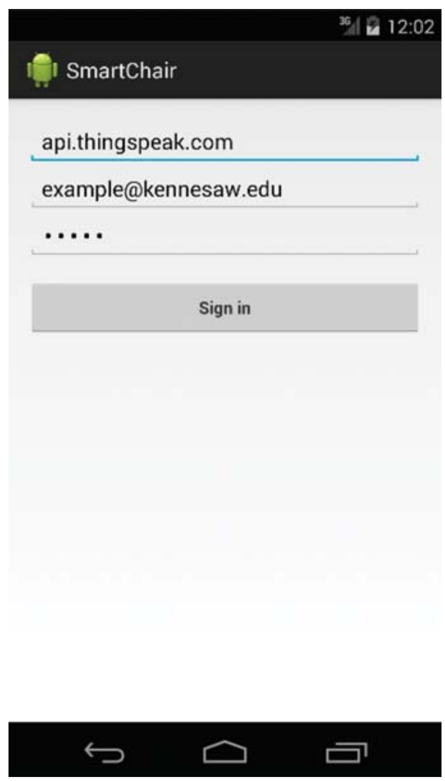
## 4. Conclusion and Future Work

The proof-of-concept experiment of the smart chair system has been completely implemented in this capstone research project. An Ardunio board based system is assembled and implemented to get user ID through RFID reader and chair occupancy status by pressure sensor. All data is uploaded to the ThingSpeak cloud. Moreover, an Android application to view the stored data has also been implemented. Through successfully implementing the project, the whole team gained invaluable knowledge in hardware design, hardware prototyping, cloud services, and android application development. After developing this project, we can recommend Ardunio hardware to those interested in implementing Internet of Things systems. Since, Ardunio hardware is easy to use and can be intergraded with numerous different modules. Moreover, the learning curve for programming on the devices is not sharp as the Ardunio developer community is large and there are many tutorials available on the Internet.

It should be noted that the possibilities exist not only in connecting the chair to internet but also in the use of real time data stored in the cloud to create new applications. This smart chair system not only offers the solution to a problem but also lays the foundation for series of future projects (such as intelligent parking system, dynamic ticketing system, *etc.*) Because of the time limitation, only the proof-of-concept




(a)




(b)

Fig. 10: (a) Registration Screen; (b) Login Screen of Mobile App.

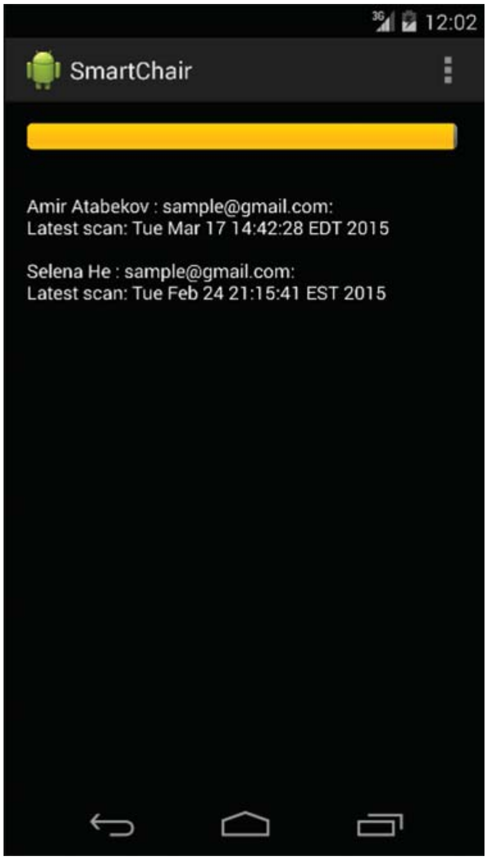



Fig. 11: Data Display Screen of Mobile App.

experiments are completed in the capstone project. Lots of additional improvements to the system can be implemented. So far only one chair connects to the internet. It is more practical to connect all chairs in a room to the internet. Hence, constructing a wireless sensor network so that every chair in a room can be connected to the internet is one research direction. Moreover, a simple mobile application is implemented in this project. Advanced application can be designed and developed, such as to show the chair map on the screen. Additionally, the occupancy status of each chair can be graphically displayed/viewed on the screen. More advanced functionality could be to let users can interact with the graph to reserve a chair. Finally, the collectted/analyzed data could be used for additional purposes, such as resource management, attendance checking, or tutor time tracking management.

## Acknowledgment

This work is funded in part by the Kennesaw State University's Office of the Vice President of Research (OVPR) Pilot/Seed Grant, by the College of Science and Mathematics Interdisciplinary Research Opportunities (IDROP) Program, and by the Department of Computer Science Mini-Research grant.

## References

[1] P.C. Blumenfeld, E. Soloway, R.W. Marx, J.S. Krajcik, M. Guzdial, and A. Palincsar, *Motivating project-based learning: Sustaining the doing, supporting the learning*, Educational Psychologist, Vol. 26, No. 3, pp. 369-398, 1991.

[2] H. Roessingh, and W. Chambers, *Project-based learning and pedagogy in teacher preparation: staking out the theoretical mid-ground*, International Journal of Teaching and Learning in Higher Education, Vol. 23, No. 1, pp. 60-71, 2011.

[3] R. C. Martin, *Agile Software Development: Principles, Patterns, and Practices*, Prentice Hall PTR Upper Saddle River, NJ, USA, 2003.

[4] S. Dorairaj, J. Noble, and P. Malik. *Knowledge Management in Distributed Agile Software Development*, Agile Conference (AGILE), pp. 64-73, Dallas, TX, Aug. 2012.

[5] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, *Internet of Things (IoT): A vision, architectural elements, and future directions*, Future Generation Computing System, 29, pp. 1645 Ű 1660, 2013.

[6] K. Ashton, *That "Internet of Things" Thing*, RFiD Journal, Vol. 22, pp. 97 Ű 114, 2009.

[7] X. Jia, O. Feng, T. Fan, and Q. Lei, *RFID technology and its applications in internet of things (IoT)*, 2nd IEEE conference on Consumer Electronics, Communications and Networks (CECNet'12), pp. 1282 - 1285, 2012.

[8] C. Sun, *Application of RFID technology for logistics on internet of things*, AASRI Conference on Computational Intelligence and Bioinformatics, pp. 106 - 111, 2012.

[9] J. He, S. Ji, Y. Pan, and Y. Li, *Constructing Load-Balanced Data Aggregation Trees in Probabilistic Wireless Sensor Networks*, IEEE Transactions on Parallel and Distributed Systems (TPDS), Vol. 25, No. 7, pp. 1681 - 1690, July, 2014.

[10] J. He , S. Ji, R. Beyah, Y. Xie, and Y. Li, *Constructing Load-Balanced Virtual Backbones in Probabilistic Wireless Sensor Networks via Multi-Objective Genetic Algorithm*, Transactions on Emerging Telecommunications Technologies (ETT), Vol. 26, No. 2, pp. 147 - 163, February, 2015.

[11] J. He, S. Ji, R. Beyah, and Z. Cai, *Minimum-sized Influential Node Set Selection for Social Networks under the Independent Cascade Model*, ACM MOBIHOC 2014, pp. 93 - 102, 2014.

[12] C. Ramos, G. Marreiros, R. Santos, and C.F. Freitas, *Smart Offices and Intelligent Decision Rooms*, pp. 851-880, Handbook of Ambient Intelligence and Smart Environments, Spriner, New York, NY, USA, 2010.

[13] J. Reijula, M. Grohn, M. Muller, and K. Reijula, *Human well-being and flowing work in an intelligent work environment*, Intelligent Buildings International, Vol. 3, Issue 4, pp. 223-237, 2011.

[14] H. Chen, *Intelligent Classroom Attendance Checking System Based on RFID and GSM*, Advanced Materials Research, Vol. 989-994, pp. 5532-5535, 2014.

[15] M. Zhi, and M. M. Singh, *RFID-Enabled Smart Attendance Management System*, Future Information Technology, Vol. 329, pp. 213-231, 2015.

[16] M. Choi, J.-H. Park, and G. Yi, *Attendance Check System and Implementation for Wi-Fi Networks Supporting Unlimited Number of Concurrent Connections*, International Journal of Distributed Sensor Networks, pp. 1 - 10, 2014.

[17] A. A. Kumbhar, K. S. Wanjara, D. H. Trivedi, A. U. Khairatkar, and D. Sharma, *Automated Attendance Monitoring System using Android Platform*, International Journal of Current Engineering and Technology, Vol. 4, No. 2, pp. 1096-1099, 2014.

[18] Ardunio Yun Board, Available online: http://arduino.cc/en/Main/ArduinoBoardYun (accessed in March 2015).

[19] ThingSpeak, Available online: https://thingspeak.com/ (accessed in March 2015).

[20] Anduino Bridge Library, Available online: http://arduino.cc/en/Reference/YunBridgeLibrary (accessed in March 2015).

[21] Arduino RFID Library for MFRC522, Available online: https://github.com/miguelbalboa/rfid (accessed in March 2015).

[22] Ardunio Serial Peripheral Interface (SPI), Available online: http://arduino.cc/en/Reference/SPI (accessed in March 2015).

# Creating Synergy between Undergraduate Senior Computing Research and Applied Capstone Courses

**M. Estep , K.D. Smith, C. Zhao, A. Johari**
Computing and Technology Department, Cameron University, Lawton, OK, USA

**W. Carney**
English Department, Cameron University, Lawton, OK, USA

**Abstract** – *This study presents a look at the synergy created between two undergraduate computing courses in the Computing and Technology Department at Cameron University – MIS 4473 Current Topics in IT, and IT 4444 IT Capstone. The focus of MIS 4473 was student group case study research, utilizing: (a) the success of Computing and Technology faculty academic publications, (b) guidelines from an international computing conference, (c) an iterative writing and editing process, and (d) a blind student peer review process. The focus of IT 4444 was applied multi-discipline systems software development in a service learning capacity for real clients. Students in MIS 4473 conducted case study research on concurrent IT 4444 capstone projects and presented findings at the end of the semester, thereby simulating the computing conference experience. The authors discuss the necessity of such synergy in relation to improved student research skills and service learning reflection.*

**Keywords:** Undergraduate Computing Research, Capstone, Software Development, Reflection, Synergy

## 1 Introduction

For a number of years, the Computing and Technology Department at Cameron University has offered the senior undergraduate course IT 4444 IT Capstone. In a service learning capacity, students from multiple disciplines of Information Technology (IT), Computer Science (CS), Multimedia (MM), and English (ENG) work together in teams to develop actual systems software for real clients. Clients benefit in gaining software for their needs and in saving hundreds of thousands of dollars not being paid to industry developers. Students benefit in gaining real software development experience and a letter of recommendation. Past clients have included, city government IT departments, city government police departments, hospitals, and others. It is common for IT 4444 students to be hired in desirable, high-paid IT positions, specifically because of their capstone experiences [1, 2]. However, Cameron University administrators overseeing campus service learning expressed a concern that a stronger reflection component may be needed in the capstone course. This has been difficult to accomplish in IT 4444, due to the rigor, scope, and scheduling of actual systems software development.

Separately, and more recently, the Computing and Technology faculty decided to add a dedicated required computing research course to the IT core curriculum. This was in accordance with the Association for Computing Machinery (ACM) IT 2008 curriculum guidelines [3] "to integrate case studies into IT courses in a way that emphasizes the importance of understanding the application domain." Initially, a temporary seminar course, MIS 4473 Current Topics in IT, was created for such a purpose. When developing this course, the instructor determined that students would benefit by conducting research according to the guidelines of an international computing conference. This was partially based on Computing and Technology faculty success in academic publication of case studies on the IT 4444 capstone class projects [1, 2]. After some research and analysis on the subject, the MIS 4473 instructor believed that students would also benefit by conducting similar group case study research on their capstone experiences, but from a student perspective. It was discovered later that MIS 4473 would also supplement the needed reflection component of the capstone course.

The remainder of this study will illustrate how synergy is created by partnership of the MIS 4473 research and IT 4444 capstone courses, from the research perspective.

# 2 Brief Literature Review

There appears to be a large number of studies in the literature focusing on undergraduate computer research separate from, and not specifically related to, applied capstone systems software development projects. However, what is important to this current study is the focus on literature regarding a possible relation of synergy between both undergraduate computing research and application.

Schneider [4] discussed a required 1-credit CS undergraduate senior capstone research seminar to simulate a computing conference experience. The seminar was taken during both semesters of the senior year. Students had some choice in selecting subject matter which was related to previous CS coursework. Activities included lectures on presenting scientific data and conducting research, Mathematica demonstrations, talks from the campus Writing Center director, and presentation techniques from a Communication Studies professor. There were also individual and group meetings with the seminar instructor used as progress checkpoints. It is important to note that the capstone mentioned here was the research course, as opposed to applied software development.

Jonas [5] described an undergraduate Computer Information Systems (CIS) capstone experience where pure research was being conducted on automated speech recognition technologies. The course had been offered each spring semester for four years as of the publishing of the study. Each year, the research was continually built upon utilizing the previous year's findings. Student grades were based upon 15% individual work, 35% group work (including a proposal and final report), and 50% anonymous and extensive student peer reviews. Possible application of research was mentioned in future plans, but not realized yet. In this study, the capstone combined research documentation and applied research in a single course.

Sun and Decker [6] discussed a dual undergraduate capstone model. One path would be structured toward conducting research for students who intend to pursue graduate studies. Students involved in this path work toward individual research mentored by faculty, with the mentor approving the research area. The other path would be structured toward software application development for students who intend to work in industry. In the second path, real non-profit clients are selected (for societal benefit). Also, it is more common for group work to transpire in the second path. This model is designed to be suitable for both CS and CIS students.

Khmelevsky [7] shared experiences of the CS department undergraduate research and teaching strategies at Massachusetts Institute of Technology. Students were able to select work on applied industry projects in multiple courses. In incorporating research into one of those courses, the purpose was to develop student academic publications. While several strategies were mentioned, there was one that stood out the most in relation to the current study - "Students need additional time to publish a research paper. In a project course students can finish an experimental part of the research project as well as an initial research, but they need extra time for research paper writing."

In summary, the author found the following undergraduate computing examples relating research and capstone in the literature: (a) the capstone as a research course to introduce students to a computing conference type of experience, (b) a capstone experience as a pure research project that documented processes for continued use in future course iteration with potential application, (c) a dual capstone experience that allowed students to choose between a research path in pursuit of attending graduate school and a path developing applications in pursuit of working in industry, and (d) a combined course that contained both a research component and an application development component, with an emphasis that more time was needed for students to conduct research.

# 3 Methodology

Based upon faculty perception of student needs and the literature, the instructor of MIS 4473 decided the best course of action would be to pattern research after the computing conference experience. The Cameron Computing and Technology faculty had experienced significant success in publishing and presenting at computing conferences. It was common for the faculty to publish case studies related to the IT 4444 capstone experience, from a faculty perspective. The MIS 4473 instructor believed students would receive similar benefit in conducting research over the capstone course experience, from a student perspective.

## 3.1 Initial Meetings

Students were informed of the purpose of MIS 4473, which was conducting research based on their concurrent IT 4444 capstone course. It was explained that IT 4444 would provide them with an applied systems software development experience, while MIS 4473 would provide them with a reflective research component. Each course would synergistically reinforce one another. IT 4444 would provide them with capstone material to conduct case study research, while MIS 4473 would help them to more fully comprehend what transpired in the capstone course.

MIS 4473 students were only required to purchase one item for the class, that being an ACM student membership with the Digital Library. The purpose was twofold. Students would have access to a vast library of computing research materials. Also, being an ACM member could be helpful in their futures as computing professionals upon graduation [8].

It has been a common experience for the Computing and Technology faculty to contribute as coauthors on published papers. To simulate this experience, MIS 4473 students were required to work in groups of no less than three and no more than five. The students were allowed the opportunity to select their own coauthors. In future assignments, students were also required to provide a detailed breakdown of each student coauthor's contribution.

## 3.2 Assignment Parameters

The WorldComp computing conference publishing guidelines were chosen as a pattern for students to follow. WorldComp is a major international computing conference with several hundred papers published and presented annually in multiple computing disciplines [9]. The MIS 4473 instructor lectured over the information found at the conference website. Students were provided with the conference website links and were required to write a detailed report on what WorldComp was about and how prospective authors engage in the publishing/presentation process. The reports were submitted through a program that checks for plagiarism, as students were required to write using their own words.

Students were also provided with faculty examples of capstone-related papers that were published in previous WorldComp conferences. The students were able to use these papers as example templates. However, the students were also reminded that these papers were written from a faculty perspective, while the students would be writing from their own perspective. A formatting template from WorldComp was also supplied to the students.

At this point, students were required to develop the beginnings of a paper title and abstract, based on their Capstone projects. It is important to mention again here that the IT 4444 capstone course had multi-discipline teams of IT, CS, MM, and ENG students, working in teams on actual systems software development for real clients. This included the development of an HTML/CSS frontend, PHP middleware, a MySQL backend database, and the use of CASE tools. Therefore, the student research groups in MIS 4473 would be focusing on different areas, depending on their involvement in IT 4444. For example, some student groups mainly focused on cyber security issues, while others focused on database programming. Student groups were also required to visit with the instructor in his office for editing and content critiques before they were allowed to submit their work. All submitted work was checked by the plagiarism program.

Several class sessions were set aside for all students to visit the Cameron University Writing Center and Library. Writing and research presentations were provided to students by the directors at both of these locations. Students were also required in future iterations of their research to utilize these facilities, as well as the ACM Digital Library.

The next step in the process was instructor graded feedback provided to each authoring group, followed by a class peer reviewed critique of the submitted title and abstract. The instructor first provided feedback upon submission. After that, all identifying information was blacked out to provide the simulation of blind peer review. The title and abstract were taken into class and projected on a screen. Then the students were collectively asked by the instructor to critique what they read, followed by instructor critique based upon the prior feedback.

Multiple iterations of writing and instructor/student peer review were performed in a similar manner. Written work was submitted to the instructor at approximately 50%, 75%, and 100% completion intervals, to coincide with work as completed in the applied IT 4444 capstone class. However, during each of these iterations, student groups were required to visit the Writing Center for critiques and editing. Afterword, they had to visit with the instructor for additional critiques and suggestions. This was again followed by graded instructor feedback. Then lastly, blind peer review (names being blacked out) took place in which the students and instructor provided critiques of the written materials projected on a screen in class.

Several class periods were also allowed for students to meet and work in their groups, as these periods provided the most efficient times for such collaboration.

Figure 1 shows two typical assignment instructrion examples…

**Required Study and Preparation: Given 1/15/2015**

- Read through the information found at…
  http://www.world-academy-of-science.org/worldcomp15/ws
  http://www.world-academy-of-science.org/worldcomp15/ws/conferences/fecs15
  http://www.world-academy-of-science.org/worldcomp15/ws/conferences/fecs15/submission
  http://world-comp.org/
  http://www.worldacademyofscience.org/worldcomp14/ws/authors/typing_instructions
- Choose your group members. Each group must have at least 3, and no more than 5, members.

**Assignment 1: Given 1/14/2015, Due 1/28/2015 by 11pm CST**

- **Exercise 1, Group:** Write a single-space, full 2-page, 12-point font, 1" margins, report on what the Worldcomp conference is about and how prospective authors engage in the process. Spelling and grammar count. Every group member must participate.
- **Exercise 2, Group:** Come up with a topic related to your major concentration area to write about. Keep in mind that the topic should have some interest and somehow present new information (it is recommended to use the Database or IT Capstone projects for this purpose). Write a title and abstract about that topic using the format from the supplied Worldcomp template and the faculty papers under *Course Documents* in Blackboard. Spelling and grammar count. The topic must be approved by the instructor approximately one week prior to submission. Points will be deducted without this approval. If deemed unsuitable by the instructor, either a revision or another topic will have to be completed. Groups must make an appointment to visit the instructor for this purpose.
- **Exercise 3, Group:** In addition, supply the name of each group member and detailed documentation of each group member's contribution.

Submit your document (in Word format) to the Assignment 1 submission link by the due dates (found in your MIS 4473 Blackboard course under *Assignments*). It is only necessary for one group member to make this submission. Also, group members will need to write using their own words, as the work will be submitted through SafeAssign. Anything beyond a 20% match may result in severe loss of points.

---

**Required Study: Given 1/28/2015**

- Ultimately, the final paper will need to be about 7 pages including all figures, tables, and references, set up according to the WorldComp conference guidelines, and with a minimum of 6 references from either .edu, .mil, .gov sources, or from published books. I have placed two papers in Course Documents that were published by our faculty at previous WorldComp conferences to give examples, along with a formatting template.
- Begin to explore the ACM website (http://www.acm.org) after you purchase your student membership with the Digital Library. There are many resources that will be of benefit to students now in a research capacity, and later in an industry capacity.
- I will arrange some class meetings for students to watch presentations given at the Academic Writing Center, and at the Library. Stay tuned for announcements on meeting dates.

**Assignment 2: Given 1/28/2015, Due 2/25/2015 by 11pm CST**

- **Exercise 1, Group: Paper Draft 1:** Using the given examples as a guide, begin putting your paper together. This will be a work in progress, approximately 50% complete, but there should also be a skeletal outline including parts that can't be completed until later. What should be more complete are areas that have taken place up to the point of the due date. Spelling and grammar count. It will be required from this point forward for each group to take their paper to the Academic Writing Center, to receive proofing before submission. It will also be required that each group will visit with me in my office, supplying documentation that the Academic Writing Center helped with proofing. I will also proof the work before submission. If any of these steps are omitted, students will receive zero credit for this assignment. I will allow several class periods to be used for groups to visit me in my office as necessary. I suggest students take advantage of this time.
- **Exercise 2, Group:** In addition, supply the name of each group member and detailed documentation of each group member's contribution.

Submit your document (in Word format) to the Assignment 2 submission link by the due dates (found in your MIS 4473 Blackboard course under *Assignments*). It is only necessary for one group member to make this submission. Also, group members will need to write using their own words, as the work will be submitted through SafeAssign. Anything beyond a 20% match may result in severe loss of points.

**Figure 1. Typical Assignment Instruction Examples**

### 3.3 Class Presentation

The final project for MIS 4473 consisted of each group presenting their findings to the class, utilizing presentation slides or other materials as necessary. As previously experienced by Computing and Technology faculty, this is exactly what transpires when a faculty paper is presented at a computing conference.

This presentation was also attended by faculty involved in teaching the IT 4444 capstone class. At the end of each presentation, questions were asked by the students, the MIS 4473 instructor, and the IT 4444 faculty. The completed papers, presentation, and question/answer period provided additional validation of utilizing the student reflection component as a supplement to the applied capstone class.

## 4 Student Perceptions

At Cameron University, students are allowed to provide anonymous comments on a standardized teacher evaluation survey each semester. At the end of the first offering of MIS 4473, there were two student comments relating to the research component…

- "The instructor was very helpful during the class. Although, the class seemed like something I could do on my own, not needing to have a structured class. I like how the instructor puts the responsibility on the student, which I enjoy…helps me to learn more."
- "I am very glad the class was changed to this research class. I just wish the class research paper was the standard for all campus research papers. This class has given me enhanced knowledge for writing and research. I only wish I could take another class like this. I believe there should be more classes that link each other like this one did with my capstone."

## 5 Discussion

The MIS 4473 instructor noticed several problematic areas students commonly encountered during the research and writing process…

- **Tense and Person:** Students would commonly mix tense and write in first or second person. The MIS 4473 instructor required students to write in past tense and third person. They had to be reminded of this on multiple occasions.
- **Long Sentences:** It was common for students to write very long sentences. Students were regularly reminded that if a sentence is read aloud, there should be natural breathing points. If one has to gasp for air, a sentence is probably too long.
- **Multiple Writers:** Students divided the work in writing sections of their papers, which is appropriate for group research. However, it was apparent in the editing process that some would not read each other's sections. The writing styles would not flow well in that case. Also, abbreviated terms were set up in an earlier section, and not used later in the paper. For example, *Information Technology (IT)* had been used early on, but *IT* was not used again. Instead, *Information Technology* was used in all later sections.
- **Native Language:** At Cameron University, a large portion of the student population speaks ENG as a second language. However, the MIS 4473 instructor would require proper ENG grammar, as it is a common language used in academic publishing. There are general patterns of writing that indicate when an author is not a native ENG speaker.
- **Citations-References:** It was common for students to express suppositions without supplying proper citations and references to support their claims. Students had to be reminded that they would not be considered experts in the field at this stage of their academic career, and that opinions alone would not be considered worthy of academic publication.
- **Procrastination:** As mentioned before, student groups had to submit successive paper iterations to the Writing Center, as well as visit with the instructor for editing and critique purposes. Several groups had a tendency to wait until a few days before the submission due date to make the required visits. In these cases, the end submissions tended to contain more of the problems mentioned above.
- **Comfort Zone:** During the MIS 4473 blind peer-review process, students were required to openly critique the group papers in class. It was common for students whose writing was critiqued to display some discomfort. In general, nobody likes to have errors critically analyzed and pointed out.

The MIS 4473 instructor believed it important for students to regularly visit the Writing Center, and separately visit the instructor, for editing and critique purposes before submitting work. Multiple iterations of this process helped students to better understand the need to avoid procrastination before paper submission. The instructor equally believed it to be important for students to participate in the class blind peer-review process. The only way for students to develop a "thick skin" during a review process is to regularly experience valid critique. In writing for conference publication, even highly experienced authors may take offense to critiques that are designed to be helpful.

It is interesting to note that during the blind peer review process in class, students progressively developed a greater sense in understanding the problems mentioned above. They also began to express a certain sense of freedom when offering valid critical review.

# 6 Conclusion

The MIS 4473 research course has now been offered twice in two years, in conjunction with the IT 4444 capstone course. In both offerings, students were given the opportunity to conduct research over the work performed in the capstone course. In the second MIS 4473 offering, two student groups indicated they would submit their papers to a future WorldComp conference. Overall, students expressed seeing positive benefits, while the faculty noticed improvement in student research skills. Also, a reflection component supplemented the capstone experience. It is the belief of the Computing and Technology faculty that such synergy would prove to be difficult without the use of separate research and applied courses working in tandem. The Computing and Technology Department is currently in the process of converting the temporary seminar course, MIS 4473 Current Topics in IT, to the permanent core course, IT 4013 Research Topics in IT.

# 7 References

[1] ***Teaching Software Engineering through a Real-World Project: A New Approach.*** Zhao, C., Estep, M., & Smith, K.D., WorldComp 2013, Proceedings of the 2013 International Conference on Software Engineering Research & Practice (SERP'13), Las Vegas, NV, July 22-25, 2013.

[2] ***Web-Based Database Project for Lawton, OK Police Department: Interdisciplinary Systems Software Development at Cameron University.*** Smith, K.D., Zhao, C., Estep, M., and Johari, A., WorldComp 2014, Proceedings of the 2014 International Conference on Frontiers in Education: CS and CE (FECS'14), Las Vegas, NV, July 21-24, 2014.

[3] ***ACM IT 2008 Curriculum Guidelines for Undergraduate Degree Programs in Information Technology,*** 8.1.2 Familiarity with application domains. Lunt, B., et al. http://www.acm.org//education/curricula/IT2008%20Curriculum.pdf.

[4] ***A New Model for a Required Senior Research Experience***. Schneider, G.M., *SIGCSE Bulletin*, Volume 34, Issue 4, December, 2002.

[5] ***Capstone Experience: Achieving Success with an Undergraduate Research Group in Speech***. Jonas, M., SIGITE '14: Proceedings of the 15th Annual Conference on Information Technology Education, Atlanta, GA, October 15-18, 2014.

[6] ***Finding an "Ideal" Model for our Capstone Experience***. Sun, N. and Decker, J., *Journal of Computing Sciences in Colleges*, Volume 20, Issue 1, October, 2004.

[7] ***Research and Teaching Strategies Integration at Post-Secondary Programs***. Khmelevsky, Y., WCCCE '11: Proceeding of the $16^{th}$ Western Canadian Conference on Computing Education, Prince George, BC, May 6-7, 2011.

[8] ***ACM Professional Membership Benefits***. http://www.acm.org/membership/benefits.

[9] ***WorldComp 2015, The 2015 World Congress in Computer Science Computer Engineering and Applied Computing***. http://www.world-academy-of-science.org/worldcomp15/ws.

# A Novel Spatial Algorithm for Similar and Non-Similar Trajectories Classification of Different Objects

**M. Elwakdy[1], M. El-Bendary[2], and M. Eltokhy[3]**
[1]Department of Electronic Technology, Helwan University, El-Amyria, Cairo, Egypt
[2]Department of Electronic Technology / Helwan University, El-Amyria, Cairo, Egypt
[3] Department of Electronic Technology / Helwan University, El-Amyria, Cairo, Egypt

**Abstract -** *In this paper, a Developed Trajectories Classification Algorithm (DTCA) of different objects: Tanker ship, Fishing boat, Deer, Cattle and Elk is presented. The points were collected of each object in the same environment. The polynomial function is used as an efficient tool to extract the features from each trajectory. The extracted features are used as input of the Subtractive Clustering, as will they are used as an input of the neural network in ANFIS.*

*There are two different types of trajectories: similar and non-similar trajectories. The similar trajectories indicate the animals' trajectories and non-similar trajectories indicate ships' trajectories. The similar trajectories of each two objects are classified together to get a high classification accuracy 98.83%, while the non-similar trajectories of ships are classified against the animals' trajectories to get a high classification accuracy 99.11% by using ANFIS. The performance of DTCA is evaluated by using different trajectories of each object.*

***Keywords:*** *Trajectories Classification of different objects; polynomial function; Subtractive clustering; ANFIS.*

## 1 Introduction

Pattern Recognition is the process to classify the input data [1] into classes based on their features. There are many applications of Pattern Recognition in radar processing, computer vision, text classification and speech recognition. Trajectories' classification [2-4] is an important research topic for predicting with the type of moving objects such as tanker ship, fishing boat, Cattle, Deer and Elk [5] based on their trajectories and other features. In previous study [6], we worked on trajectories' classification of two different types of ships (Fishing boat and Tanker ship) using a Polynomial Function, subtractive clustering and ANFIS and got high classification accuracy "99.5 %" and average error rate "0.0051065".

In this research work, we worked on predicting with the type of the object (fishing boat, tanker ship, Elk, cattle and Deer) which a trajectory belongs to. TCA [6] is developed to have the ability to classify the similar/non-similar trajectories of different types of objects. The computer experiments revealed that DTCA has the ability to discriminate between the similar trajectories of the different types of objects like Elk, Deer and Cattle, as it has the ability to classify the trajectories of different types of objects (Tanker ship & Elk, Fishing boat & Elk, etc.) which have non-similar trajectories. This presented innovative method can be used widely in the future for discriminating between the different trajectories of different objects regardless what the type of object is.

ANFIS is used to enhance the prediction capabilities where it combines the advantages of both adaptive neural networks and the fuzzy logics qualitative approach in a single framework [7, 8] where this makes a fuzzy system less depending on expert systems and more methodical [9]. The performance of Artificial Neural Network (ANN) is poor if the size of training data is small [10, 11]. In this study, ANFIS is used for the trajectories' classification of different objects. The training and checking data are prepared well by partitioned all trajectories of all objects to many segments and then using the polynomial function for extracting the coefficients from all segments before uploading them to ANFIS Editor to ensure a good representation of all trajectories.

This paper is organized as follows: Section 2 states the trajectories characterization of each object. Section 3 states the features extraction using the polynomial function, subtractive clustering and ANFIS. Section 4 states training parameters using the ANFIS. The efficiency of the proposed method in classifying all trajectories of all objects using ANFIS is demonstrated in Section 4. In Section 5, the related works are discussed. In Section 6, conclusions are presented. Finally, references are presented in Section 7.

## 2 Trajectories

The word "trajectory" means the path that the moving object follows as a function in time. There are different trajectories of different objects (tanker ship, fishing boat, Elk, Cattle and Deer) are used for the classification purpose. The trajectory can be described mathematically either by the position of each object over time or by the geometry of the path. In this study, all trajectories of all objects are represented as a sequence of 2-dimensional points (vectors or trajectory elements) [4, 5] <$x_1$; $y_1$; $t_1$>; …; < $x_n$; $y_n$; $t_n$> where $x_n$ and $y_n$ represent the position of each object at time $t_n$. In this paper, each trajectory of both objects is presented as T = < $x_1$; $y_1$ >; …; < $x_n$; $y_n$ > because the temporal dimension is discarded. The sample rate of all trajectories is not constant where the temporal difference between the sequential samples is not always the same. The trajectories of all objects are different in temporal length, the number of data points and distance traveled. All trajectories of all objects are partitioned to many line segments [2, 3, 5, 12], and the used real trajectories database [5] includes the trajectories of all objects which are used in this study. The number of points of all trajectories of each object is equal, as the number of the trajectories of all objects, which used for checking and training, is equal as shown in Table. I.

TABLE. I. Number of Trajectories of Each Object (Points)

| Type of Object | Number of Trajectories (points) |
|---|---|
| Tanker ship | 6 (530) |
| Fishing Boat | 6 (896) |
| Elk | 6 (460) |
| Cattle | 6 (248) |
| Deer | 6 (244) |

## 3 Proposed Trajectories Classification Algorithm (TCA)

The DTCA consists of three parts that are: features extraction, subtractive clustering and ANFIS. The proposed algorithm is used to carry the goal of this research.

The proposed algorithm is illustrated in Fig.1.

The following subsections are explained the above system's block diagram.

### 3.1 Features Extraction

Polynomial is a smooth and continuous function [13] which is used for extracting the coefficients from all trajectories of all objects where all trajectories are partitioned to many segments. All segments are fitted by using a polynomial function [14]. The polyfit function in Matlab

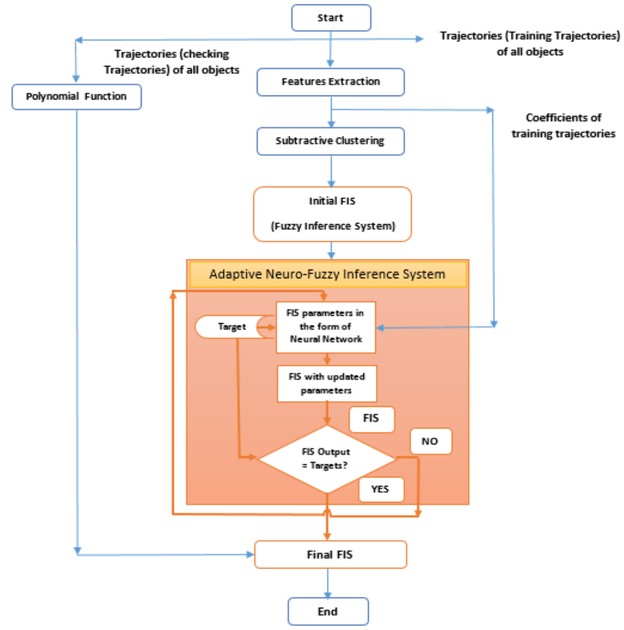
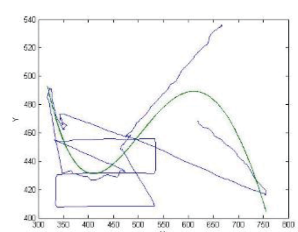



Fig.1. Proposed Algorithms of the Trajectories' Classification (DTCA) of different objects using Features Extraction, Subtractive Clustering and ANFIS.

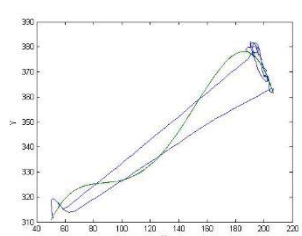
"polyfit ($x$, $y$, $n$)" [15] where '$x$' and '$y$' indicate the number of points of each segment of all trajectories and '$n$' indicates the degree of the polynomial function. This function is used for returning the coefficients for a polynomial $p(x)$ of degree $n$ of all segments of all trajectories for getting a best fit of trajectory data in $y$. A polynomial function with a degree $n$ is represented as in Eq. (1):

$$p(x) = p_1\, x^n + \; p_2\, x^{n-1} + \cdots + p_n x + p_{n+1} \qquad (1)$$

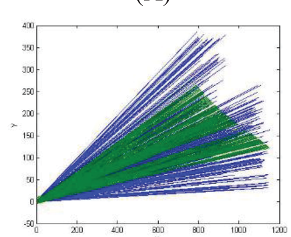
Where $p_1$, $p_2$,…, $p_{n+1}$ indicate the coefficients for the polynomial function. The coefficients don't represent the similar and non-similar trajectories as shown in Fig.2.



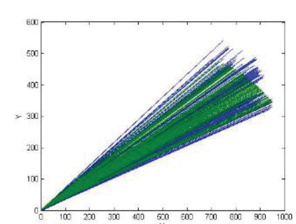
(A)



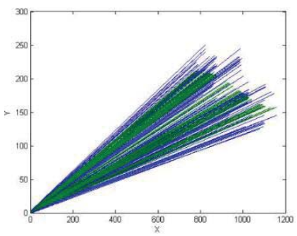
(B)



(C)



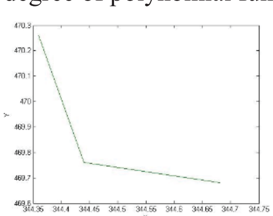
(D)



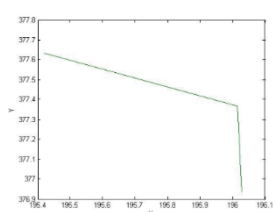
(E)

Fig.2. (A) Blue line indicates the fishing boat trajectory at sea. The Green line depicts to plot Y through the constructed polynomial model evaluated at each X- component values; (B) Blue line indicates the tanker ship trajectory at sea. The Green line depicts to plot Y through the constructed polynomial model evaluated at each X- component values; (C) Blue line indicates the Elk trajectory. The Green line depicts to plot Y through the constructed polynomial model evaluated at each X- component values; (D) Blue line indicates the Cattle trajectory. The Green line depicts to plot Y through the constructed polynomial model evaluated at each X- component values; (E) Blue line indicates the Deer trajectory. The Green line depicts to plot Y through the constructed polynomial model evaluated at each X- component values.

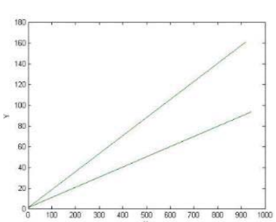
All trajectories of all objects are partitioned to many segments for getting a good representation of all trajectories of all objects as shown in Fig.3. All segments have the same number of points (three sequential points) as much as possible. By this way, we got a good representation of all trajectories of all objects. The number of segments depends on the trajectories length where the trajectories length, which indicates the number of points of all trajectories of all objects, is not constant because this depends on the type of object. In other words, the number of points of all trajectories of each object is equal. The segments number is related with the degree of polynomial function. This means that if the number of segments is increased, the degree of polynomial function will be decreased where the number of coefficients, which need to represent each segment, will be decreased. After a lot of experiments, the degree of polynomial function is used ‘4’.



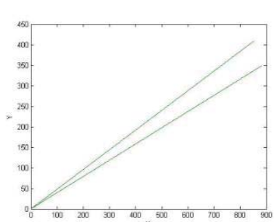
(A)



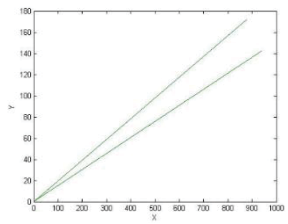
(B)



(C)



(D)



(E)

Fig.3. (A) Good representation of a segment is shown by using a polynomial function. The segment of fishing boat trajectory consists of three consecutive points; (B) Good representation of a segment is shown by using a polynomial function. The segment of tanker ship trajectory consists of three consecutive points; (C) Good representation of a segment is shown by using a polynomial function. The segment of Elk trajectory consists of three consecutive points; (D) Good representation of a segment is shown by using a polynomial function. The segment of Cattle trajectory consists of three consecutive points; (E) Good representation of a segment is shown by using a polynomial function. The segment of Deer trajectory consists of three consecutive points.

In this study, the total number of training and checking trajectories of each object is 6. 50 % of the trajectories of each object is used for training and 50 % of the trajectories of each object is used for checking by ANN. The coefficients of all segments of training and checking trajectories of each object (Tanker ship, Fishing boat, Cattle, Elk or Deer) are divided to 11 groups. The total number of groups (5 objects x 11 groups of each object) is 55. The number of coefficients of each group depends on the number of the extracted coefficients from all segments of all trajectories of each object. The ‘standard deviation’, ‘mean value’, max function’ and ‘bsxfun function’ in MATLAB [16, 17] are used for make a good preparation of training and checking datasets, which include all coefficients of all segments of all trajectories of both objects, and reduce the size of training and checking datasets before the Subtractive Clustering stage as shown in Fig. 4.

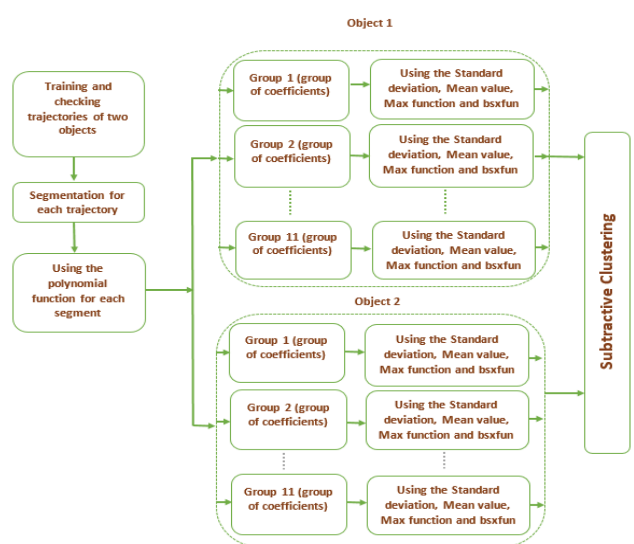



Fig.4. Analysis Framework

The proposed algorithm (TCA) [6] is developed by using these functions (‘standard deviation’, ‘mean value’, max function’ and ‘bsxfun function’) to have a strong ability to classify different trajectories regardless the type of object and get highest classification accuracy with lowest average error.

### 3.2 Subtractive Clustering

The subtractive Clustering is proposed by Chiu as an efficient fuzzy clustering algorithm [18]. The subtractive clustering is used for find the cluster centers which specify through the data points themselves, as it is used to specify the number of clusters through the training data. Before the Subtractive Clustering Stage, the coefficients of all trajectories of all objects are prepared for the classification purpose. Consider a group of 22 data points ‘$n$’ of each two objects $\{z_1, z_2, z_3, \ldots, z_n = 22\}$ [19] where $z_i$ is a vector in the feature space. The feature space is assumed that it is normalized where all data points are bounded by a unit hypercube. Through the Subtractive Clustering, the data point, which have a high potential value, is computed relied on the density of surrounding data points and this point is more appropriated to be the potential cluster center. Each data point supposes that it may be a potential cluster center. Therefore, the initial potential value is associated to $z_i$ [19] as in Eq. (2):

$$P_i = \sum_{j=1}^{n} e^{-\alpha(z_i - z_j)} \qquad (2)$$

α=4/$r_a^2$ where $r_a^2$ is a positive constant ($r_a$>0) defining a neighborhood radius of each point. Because the points $z_j$ located out the radius of $z_i$, points $z_j$ will have small effect in its potential. Only the points, which are near to $z_i$, will have high potential values.

After the potential values are calculated to each point, the one with highest potential value will be nominated to be the first cluster center, and the potential of all the remaining points is revised. If $z_1^*$ is defined as a first cluster center with potential value $p_1^*$, the potential of all the existing points is revised as in Eq. (3):

$$p_i = p_i - p_1^* e^{-\beta(z_i - z_1^*)^2} \qquad (3)$$

$$i = 1, 2, \ldots, N \ ; \ i \neq j$$

β=4/$r_b^2$ where $r_b$ >1 defines as a neighborhood radius with significant potential reductions. The potentials of all points, which near the nominated center, will be greatly reduced so the probability to select these points to be the cluster center will be decreased. This prevents the concentration of corresponding clusters in denser zones. Therefore, $r_b$>$r_a$ ($r_b$=1.5$r_a$) to hide closely spaced clusters. After the potential of these points is reduced, the one with the highest potential is nominated as a second cluster then the potentials of the remaining points will be reduced again. The potential is revised after the $k_{th}$ group is determined [19] as in Eq. (4):

$$p_i = p_i - p_k^* e^{-\beta(z_i - z_k^*)^2} \qquad (4)$$

Where $z_k^*$ indicates the location of $k_{th}$ cluster center and $p_k^*$ is its potential value. The process will continue until the shopping criterion defined in [20] that is reached.

There are 4 clustering parameters: Range of Influence, Squash Factor, Accept Ratio and Reject Ratio that are used in Subtractive Clustering stage. These parameters play important role in determining the clusters specifications. There are a lot of experiments that we did to get a best representation of the training data by using the Subtractive Clustering. At the end of clustering, the Initial Fuzzy Inference System (IFIS) is generated with a group of rules and membership functions [21]. The center of each membership function indicates the center of each cluster, as the width of each membership function obtained on the basis of the radius [22].

The Accept and reject ratios are important parameters in determining the cluster center [19], as the value of Range of Influence parameter, which determines the clusters radius, plays an important role in determining the number of clusters. When the clusters radius is a small value (less than 0.1), the number of clusters, membership functions and rules will be increased. When the cluster radius is a big value (less than 1 and more than or equal 0.1), the number of clusters, membership functions and rules will be decreased. This means that this parameter controls strongly in the number of clusters, number of membership function and rules.

The trajectories of the Cattle and Deer are very similar if they are compared with the Elk’s trajectories. The clustering parameters play important role in classifying the similar trajectories (Animals’ Trajectories), but the value of Range of Influence parameter makes a significant effect in the classification process. If the trajectories are similar, the value of Range of Influence parameter can be small and/or big value. After a lot of experiments that we did, put a small value of Range of Influence parameter that is preferred to ensure a higher classification accuracy of all trajectories and best representation of the training data of both objects. When the value of Range of Influence parameter is a small value, this will perform to increase the clusters number for getting best representation of the training data of both objects which will reflect on the number of membership functions and rules as well. The clustering parameters, which are used to classify the similar trajectories of different animals, are the same.

On the other hand, the tanker ship’s trajectories (simple trajectories) and fishing boat’s trajectories (bending trajectories) are classified against animals’ trajectories by using ANFIS. In this case, the trajectories of both objects are non-similar. The value of Range of Influence parameter can be big and/or small value depends on the trajectories of the two objects which are classified. When Tanker ship’s trajectories and fishing boat’s trajectories are classified against the animals’ trajectories, a small value of Range of Influence parameter is preferred to ensure high classification accuracy of all trajectories and best representation of the training data of both objects. Generally, the clustering parameters are the same, in case; tanker ship's trajectories are classified against the trajectories of Cattle, Deer or Elk because the tanker ship's trajectories are simple, as the tanker ship moves in limited trajectories whether they compared with fishing boat. On the other hand, the clustering parameters are changed (not constant), in case; fishing boat's trajectories are classified against the trajectories of Cattle, Deer or Elk because the fishing ship's trajectories are bending, as fishing boat moves in non-limited trajectories so the values of clustering parameters are not the same when the fishing boat's trajectories are classified against animals' trajectories.

The change of clustering parameters based on the objects' trajectories is explained in the following cases:

### 3.2.1 Case 1: Similar Trajectories

The clustering parameters, which prove instrumental in ANFIS training, are used for the animals' trajectories (Elk & Cattle, Elk & Deer and Cattle & Deer) that are shown in Table II.

TABLE. II. Clustering Parameters of the "Elk and Cattle trajectories", "Elk and Deer trajectories" and "Cattle and Deer trajectories"

| Clustering Parameters | Elk - Cattle | Elk - Deer | Cattle - Deer |
|---|---|---|---|
| Range of Influence | 0.03 | 0.03 | 0.03 |
| Squash Factor | 1.25 | 1.25 | 1.25 |
| Accept Ratio | 0.5 | 0.5 | 0.5 |
| Reject Ratio | 0.15 | 0.15 | 0.15 |

### Case 2: Non-Similar Trajectories

The clustering parameters are used for the non-similar trajectories (Tanker Ship & Cattle, Tanker Ship & Deer, Tanker Ship & Elk, Fishing boat & Cattle, Fishing boat & Deer and Fishing boat & Elk) that are shown in Table III, IV.

TABLE. III. Clustering Parameters of the "Tanker ship and Cattle trajectories", "Tanker ship and Deer trajectories" and "Tanker ship and Elk trajectories"

| Clustering Parameters | Tanker Ship - Cattle | Tanker Ship - Deer | Tanker Ship - ELK |
|---|---|---|---|
| Range of Influence | 0.04 | 0.04 | 0.04 |
| Squash Factor | 1.25 | 1.25 | 1.25 |
| Accept Ratio | 0.5 | 0.5 | 0.5 |
| Reject Ratio | 0.1 | 0.1 | 0.1 |

TABLE. IV. Clustering Parameters of the "Fishing boat and Cattle trajectories", "Fishing boat and Deer trajectories" and "Fishing boat and Elk trajectories"

| Clustering Parameters | Fishing Boat - Cattle | Fishing Boat - Deer | Fishing Boat - ELK |
|---|---|---|---|
| Range of Influence | 0.03 | 0.07 | 0.06 |
| Squash Factor | 1.25 | 1.25 | 1.25 |
| Accept Ratio | 0.5 | 0.5 | 0.5 |
| Reject Ratio | 0.1 | 0.1 | 0.1 |

## 3.3 Adaptive Neuro-Fuzzy Inference System (ANFIS)

The Adaptive Neuro-Fuzzy Inference System technique was presented by Jang in 1993 [8]. ANFIS is used in many applications [7, 23-28] where it is a very powerful learning system. The ANFIS is widely used in a lot of scientific and engineering applications. There are many advantages of ANFIS which include in describing the behavior of the complicated systems by refining the fuzzy IF-THEN rules, it is simple to implement, it doesn't not require previous human expertise and it is fast and precise learning [29, 30]. In this study, ANFIS is used to classify the trajectories of different objects based on the extracted features. The IFIS is trained by using the ANN to get a Final FIS where the least possible error between the desired output and FIS output is obtained as shown in Fig.1.

ANFIS is used to adjust (train) the input and output membership parameters for reaching to optimal distribution of membership functions [31] through neural network's learning algorithm (hybrid-learning algorithm) [8] [32,33] using input and output training data (Trajectories data). The hybrid-learning algorithm combines the back-propagation algorithm and Least Squares algorithm [7] which used for training the Fuzzy Inference System [34]. The Hybrid FIS is trained in 3 Epochs, and the Error tolerance is kept zero for the process. The average error rate and classification accuracy are calculated by figuring out the difference between the FIS output curve and checking data curve [7].

In Fig.5, basic structure of ANFIS includes two inputs $x$ and $y$ and one output $fi$ [8, 30-35]. ANFIS's architecture includes five layers. All nodes in each layer are described by the nodes functions. Each node has the similar function in the same layer. The layers: 1 and 5, which represent training values and the predicted values, indicate the input and output nodes. The square nodes, which are named adaptive nodes, indicate the parameters are adjustable in these nodes while the circle nodes, which are named fixed nodes, indicate the parameters are fixed (not adjustable).

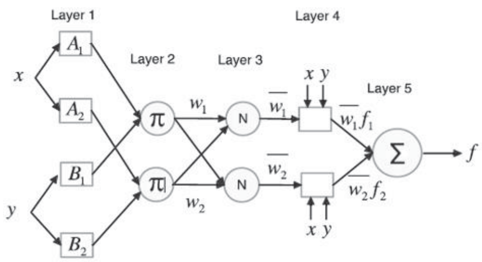



Fig.5. ANFIS architecture [8].

To present the ANFIS's architecture, two Takagi-Sugeno IF-THEN rules are considered as follows:

*Rule 1*: *If x is* $A_1$ *and y is* $B_1$ *Then* $f_1 = p_1x + q_1y + r_1$
*Rule 2: If x is* $A_2$ *and y is* $B_2$ *Then* $f_2 = p_2x + q_2y + r_2$

Where $Ai$ and $Bi$ are the fuzzy sets and $p_i$,$q_i$ and $r_i$ are the design parameters which determined during the training process. The node function in each layer is explained in detail below:

**Layer 1**: This layer is fuzzification layer where all the nodes are adaptive nodes and each node, which represents a membership value in this layer, is an adjustable node with node function. The outputs of this layer is a fuzzy membership grade of the inputs *x*, *y* are given by Eq. (5), (6):

$$O_{1,i} = \mu Ai(x), i = 1,2, \qquad (5)$$

$$O_{1,i} = \mu Bi - 2(y), i = 3,4, \qquad (6)$$

Where $Ai$ and $Bi$ are the linguistic labels (high, low, etc.) which are related with this node function. $\mu Ai(x)$, and $\mu Bi - 2(y)$ can adopt any fuzzy membership function. If $\mu Ai(x)$ is a Gaussian membership function, $\mu Ai(x)$ is given by Eq. (7):

$$\mu Ai(x) = exp[-\left(\frac{x - C_i}{a_i}\right)^2] \qquad (7)$$

Where $a_i$ and $C_i$ are parameters of the membership functions.

**Layer 2**: The nodes in this layer are fixed nodes. The output of each node represents the firing strength of the rule. The nodes in this layer are labeled with $\pi$ and they play as simple multipliers. The output of this layer is given by Eq. (8):

$$O_{2,i} = \omega_i = \mu Ai(x) * \mu Bi(y), i = 1,2. \qquad (8)$$

Where $\omega_i$ indicates the firing strength of the rule.

**Layer 3**: The nodes in this layer are fixed nodes. In this layer, each node calculates the ratio of the rule's firing strength to the total of all rules' firing strengths. The nodes in this layer are labeled with N. The output of this layer '$O_{3,i}$' is called normalized firing strength and is given by Eq. (9):

$$O_{3,i} = \varpi_i = \frac{\omega_i}{\omega_{1+}\,\omega_2}, i = 1,2. \qquad (9)$$

**Layer 4**: The nodes in this layer are adaptive nodes. The output of each node is the product of the normalized firing strength and the first order polynomial (the function of consequent). The output of this layer is given by Eq. (10):

$$O_{4,i} = \varpi_i f_i = \varpi_i\,(p_i x + q_i y + r_i), i = 1,2, \qquad (10)$$

Where $\varpi_i$ indicates the output of layer 3, and $p_i$,$q_i$ and $r_i$ are the consequent parameters.

**Layer 5**: The node in this layer is a fixed node. The node in this layer is labeled with $\sum$. In this layer, the overall output is calculated as a summation of all inputs by Eq. (11):

$$O_{5,i} = \sum_i \varpi f_i = \frac{\sum_i \omega_i f_i}{\sum_i \omega_i} \qquad (11)$$

After Final FIS is generated, ANFIS is evaluated by the checking data where the checking data is compared with the trained data. The total number of IF-Then rules is changed depends on the trajectories of the two objects which are classified by using the ANFIS and clustering parameters. Each IF-then rule consists of six coefficients: five coefficients are multiplied with five inputs in addition to the constant. The IF-Then rules are the essence of fuzzy logic [7] where they demonstrate the relation between the input and output variables [25].

In this study, the number of membership functions is changed based on the trajectories of the two objects, which are classified by using ANFIS, and the clustering parameters.

# 4 Experiment Results

In this study, the Sugeno ANFIS is implemented in MATLAB R2011a. The training and checking data are used for evaluating the proposed algorithm. The trajectories of different objects: Elk, Deer, Cattle, fishing boat and tanker ship are classified by using ANFIS. The total number of all trajectories of all objects (5 objects x 6 trajectories of each object) is 30. The trajectories of each two objects are classified to appear the ability of proposed algorithm in discriminating between the trajectories of different objects regardless the type of object. Six trajectories (three trajectories of each object) are used as a training data, and the other six trajectories (three trajectories of each object) are used as a checking data for the proposed algorithm.

After the IFIS is trained by using ANN, the trained FIS is tested against the training and checking data to observe the difference in average testing error between both of them. IFIS is trained many times to get lowest average error and highest classification accuracy which are changed based on the trajectories of objects (similar and non-similar trajectories) are classified together by using ANFIS. When similar trajectories of different objects (Elk, Deer and Cattle) are classified against each other, the highest classification accuracy is 98.83% and lowest average error is 0.011736. When the non-similar trajectories of the fishing boat and tanker ship are classified against the animals' trajectories, the highest classification accuracy is 99.11 % and lowest average error is 0.0089393.

The average testing error is calculated by testing the training and checking data of different trajectories of each two objects against the trained FIS are explained in the following cases:

## 4.1 Case 1: Similar Trajectories

The average testing error is calculated by testing the training and checking data of Elk and Cattle trajectories against the trained FIS output as shown in Table V.

TABLE. V. The average testing error of the training and checking data "Elk and Cattle trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 2.7542e-6 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.011736 | 98.83 |

The average testing error is calculated by testing the training and checking data of Elk and Deer trajectories against the trained FIS output as shown in Table VI.

TABLE. VI. The average testing error of the training and checking data "Elk and Deer trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 6.8083e-6 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.014409 | 98.6 |

The average testing error is calculated by testing the training and checking data of Cattle and Deer trajectories against the trained FIS output as shown in Table VII.

TABLE. VII. The average testing error of the training and checking data "Cattle and Deer trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 6.6341e-6 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.021961 | 97.8 |

## 4.2 Case 2: Non-Similar Trajectories

The average testing error is calculated by testing the training and checking data of Tanker ship and Elk trajectories against the trained FIS output as shown in Table VIII.

TABLE.VIII. The average testing error of the training and checking data "Tanker ship and Elk trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 2.4406e-6 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.11024 | 88.98≈ 89 |

The average testing error is calculated by testing the training and checking data of Tanker ship and Cattle trajectories against the trained FIS output as shown in Table IX.

TABLE. IX. The average testing error of the training and checking data "Tanker ship and Cattle trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 3.1873e-7 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.018707 | 98.13 |

The average testing error is calculated by testing the training and checking data of Tanker ship and Deer trajectories against the trained FIS output as shown in Table X.

TABLE. X. The average testing error of the training and checking data "Tanker ship and Deer trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 7.2251e-7 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.053301 | 94.7 |

The average testing error is calculated by testing the training and checking data of Fishing boat and Elk trajectories against the trained FIS output as shown in Table XI.

TABLE. XI. The average testing error of the training and checking data "Fishing boat and Elk trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 0.00018516 | 99.98 |
| Testing the checking data against the trained FIS | 0.0089393 | 99.11 |

The average testing error is calculated by testing the training and checking data of Fishing boat and Cattle trajectories against the trained FIS output as shown in Table XII.

TABLE. XII. The average testing error of the training and checking data "Fishing boat and Cattle"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 3.263e-7 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.018679 | 98.1 |

The average testing error is calculated by testing the training and checking data of Fishing boat and Deer trajectories against the trained FIS output as shown in Table XIII.

TABLE. XIII. The average testing error of the training and checking data "Fishing boat and Deer trajectories"

| The average testing error | Value | Accuracy (%) |
|---|---|---|
| Testing the training data against the trained FIS | 4.4746e-6 | More than 99.99 |
| Testing the checking data against the trained FIS | 0.074412 | 92.6 |

# 5 Related Works

There are many applications (or methods) which have been proposed in the fields of pattern recognition [36] such as bioengineering [37], data mining, video surveillance [38, 39], computational linguistics, biometrics, bioinformatics and etc. In some proposed methods, the Hidden Markov Model (HMM) and statistical approaches are used to classify the data into different categories (classes or clusters). G. Vries, W. Hage and M. Someren [4] worked on predicting with behavior of the group of vessels, which use the shipping lane to move to the north, and discriminating between them based on the trajectories belongs to. In this paper, the Authors used kernel based algorithms for clustering, where the vessels' trajectories put into groups of similar movement patterns, and classification using the Support Vector Machine (SVM) classifier. For the purpose of getting a best average accuracy, the Authors combined the low trajectory information and geographical domain knowledge. The best average classification accuracy, that they got, is 75.4%.

J. Lee, J. Han, X. Li and H. Gonzalez [5] worked on trajectories' classification of moving objects: Fishing boot, Tanker ship, Elk, cattle and Deer based on their trajectories. There are huge amounts of trajectory data that were collected of each object. The real and synthetic datasets, which include animals' movement data and ships' positioning data, are used for the purpose of trajectories' classification between different objects and got a high classification accuracy. There are many scientists and researches that are interested in trajectories' classification based on the shapes of whole trajectories, but this preformed that the extracted features didn't represent the whole trajectories. In this paper [5], the Authors worked on generating a hierarchy of features of all trajectories of all objects by dividing the trajectories of all objects to many segments (sub-trajectories) and exploring two types of clustering: region-based and trajectory-based, which perform to get a high classification accuracy when the two types of clustering are collaborated together, before using the SVM classifier. When the non-similar trajectories (trajectories of two different kinds of ships) are classified by using SVM classifier, the high classification accuracy is 98.2 % where the two types of clustering are collaborated together. When similar trajectories (Animals' trajectories) are classified by using SVM classifier, the high classification accuracy is 83.3% where the two types of clustering are collaborated together.

R. Pelot and Y. Wu [2] worked on analyzing the spatial patterns of recreational boats: canoes, kayaks, motorboats and sailboats and classified the trajectories of all of them based on the difference in the movement characteristics. There are some factors that may affect the trajectories characteristics of recreational boats like weather conditions, geographical locations and operator traits. The GPS data is accumulated from two selected areas to know whether the geographic locations affect the trajectories of recreational boats. These two areas are coastal area which is close to Halifax and Saint John River area which is close to Fredericton. The trajectories of all recreational boats were recorded by using GPS receiver which helped on extracting some other features from the boating trajectories like Mean Speed (MS), Max1/20 Speed, Mean Turning Angle (MTA), Aspect Ratio (AR), Coverage Index (CI), Total Distance Traveled (TD) and furthest distance from shore (DFS). These variables are used to know the type of recreational boat, but MS, MTA and DFS are most active variables in distinguishing between the boat types. The movement characteristics of each boat are grouped separately to know whether their trajectories are different. The statistical multivariate approach is used to determine the group to which an object belongs by using a forming discriminate function of each boat group. Based on the forming discriminate functions, the highest classification accuracy in distinguishing between the boat types is 99.7%. The trajectories of Canoes and kayaks are similar where they are manually propelled while the motorboats and sailboats, which are driven by the force of wind, have distinct trajectories on Canoes and kayaks trajectories. In coastal area, the highest classification accuracy for discriminating between the similar trajectories of Canoes and kayaks is 33.3 %. In river area, the highest classification accuracy for discriminating between the similar trajectories of Canoes and kayaks is 25%.

We worked on extracting the coefficients from all segments of all trajectories of different kinds of objects: Tanker ship, Fishing boat, Elk, Cattle and Deer using the polynomial function. The used databases [5] contain animals' trajectories (Elk, Deer and Cattle) and ships' trajectories (tanker ship and fishing boat). After the coefficients are extracted from all segments of all trajectories, the training and checking datasets of all coefficients of all trajectories of all objects are prepared well where the coefficients of each two objects are prepared in two matrices (training and checking matrices) before uploading them to the ANFIS for the classification purpose. The classification accuracy of trajectories is not constant where it depends on the trajectories of two objects, which are classified together, and clustering parameters. There are two different kinds of trajectories: similar trajectories, which represent the animals' trajectories, and non-similar trajectories, which represent the ships' trajectories. In this study, we could obtain highest classification accuracy (98.83%) in classifying the similar trajectories compared with [2] and [5] where the trajectories of each two objects (two animals) are classified by using ANFIS, as we could obtain highest classification accuracy (99.11%) with compared with [4] and [5] where the tanker ship's trajectories and fishing boat's trajectories against animals' trajectories are classified by using ANFIS. The fishing boat's trajectories and animals' trajectories which were used in this study and previous study as in [5] are not simple trajectories whether they compared with the recreational boats' trajectories as in [2]. The innovative method is a simple method in discriminating between the different trajectories of different objects and get a high classification accuracy whether it compared with the other methods as in [5].

# 6 Conclusions

In this paper, a novel method for classifying the trajectories of different objects (Fishing boat, Tanker ship, Cattle, Deer and Elk) has been presented. The polynomial function is used as an effective tool to extract the coefficients from all segment of all trajectories of all objects. The subtractive clustering is an efficient tool to put the training data in group of clusters. The clusters number and cluster centers are determined by the clustering parameters. IFIS, which is generated by the subtractive clustering, is trained by the ANN for getting a Final FIS (trained FIS). The training and checking data are tested against the trained FIS to evaluate the performance of DTCA.

There are two different types of trajectories: similar and non-similar trajectories. The animals' trajectories are similar and this performed to find some difficulties in discriminating between them and got high classification accuracy. In this study, the trajectories of all animals are classified against each other by using ANFIS and got a high classification accuracy (98.83%). The ships' trajectories are classified against the animals' trajectories by using ANFIS and got a high classification accuracy (99.11%).

This study proves the ability of DTCA in classifying the different trajectories of different objects effectively, where the trajectories of each two object are classified together, and get highest classification accuracy and lowest average error, which is close to the predefined tolerance limit, regardless what the type of object is.

# 7 References

[1] C. Bishop. "Pattern Recognition and Machine Learning"; Springer -Verlag New York, ISBN 978-0-387-31073-2, 2006.

[2] R. Pelot, Y. Wu. "Classification of recreational boat types based on trajectory patterns"; ScienceDirect, Vol. 28, Issue 15, pp. 1987–1994, 2007.

[3] J. Lee, J. Han, K. Whang. "Trajectory Clustering: A Partition-and-Group Framework"; Proceedings of the ACM SIGMOD international conference on Management of data, pp. 593-604, 2007.

[4] G. Vries, W. Hage, M. Someren. "Comparing Vessel Trajectories using Geographical Domain Knowledge and Alignments"; Data Mining Workshops (ICDMW), IEEE International Conference, Sydney, NSW, pp. 209-216, 2010.

[5] J. Lee, J. Han, X. Li, H. Gonzalez. "TraClass: Trajectory Classification Using Hierarchical Region-Based and Trajectory-Based Clustering"; Proceedings of the VLDB Endowment, Vol. 1, Issue 1, pp. 1081-1094, 2008.

[6] M. Elwakdy, M. El-Bendary, M. Eltokhy. " A Novel Trajectories Classification Approach for different types of ships using a Polynomial Function and ANFIS"; Unpublished conference paper, IPCV'15 - The 19th International Conference on Image Processing, Computer Vision, & Pattern Recognition, July 27-30, Las Vegas, USA, 2015.

[7] A. Abbas, A. Mazhar, S. Hassan. "Measuring Weather Prediction Accuracy Using Sugeno Based Adaptive Neuro Fuzzy Inference System, Grid Partitioning and Guassmf"; Computing Technology and Information Management (ICCM), 8th International Conference, pp. 214 – 219, 2012.

[8] R.-S, R. Jang. "ANFIS: Adaptive-Network-based Fuzzy Inference Systems"; IEEE Trans. Svst., Man, Cybern , Vol.23, Issue 3, pp. 665-684, 1993.

[9] H. Cao, G. Si, Y. Zhang, X. Ma. "A hybrid controller of self-optimizing algorithm and ANFIS for ball mill pulverizing system"; IEEE International Conference on Mechatronics and Automation, pp. 3289-3294, 2007.

[10]E. Visser, M. Otsuka, T.Lee "A spatio-temporal speech enhancement scheme for robust speech recognition in noisy environments"; Ingentaconnect, Speech Communication, Vol.41, Issue 2, pp. 393-407, 2003.

[11]P. Nava, J. Taylor. "The Optimization of Neural Network Performance through Incorporation of Fuzzy Theory"; Proceedings of the Eleventh International Conference on Systems Engineering, pp. 897-901.

[12]H. Li, X. Wang. "Automatic Recognition of Ship Types from Infrared Images Using Support Vector Machines"; Computer Science and Software Engineering, International Conference, Wuhan, Hubei, Vol. 6, pp. 483 – 486, 2008.

[13]Z. Shan, Y. Yang. “Learning Polynomial Function Based Neutral-Emotion GMM Transformation for Emotional Speaker Recognition”; Pattern Recognition, 19th International Conference on, ICPR 2008, pp. 1-4, 2008.

[14]L. Li, B. Zhang , L. Gao, Y. Wu. “Spectral Feature Extraction Based on Orthogonal Polynomial Function Fitting”; Pattern Recognition in Remote Sensing (PRRS), 8th IAPR Workshop on, pp.1-4, 2014.

[15]”Polynomial Function,” Mathimatics Toolbox, MathWorks, Matlab R2011a.

[16]”mathematics”, Statistics and Random Numbers, Descriptive Statistics, MathWorks, Matlab R2011a.

[17]”Mathematics”, Linear Algebra, Matrix Functions, MathWorks, Matlab R2011a.

[18]S. Chiu. "Fuzzy Model Identification Based On Cluster Estimation"; J. of Intelligent and Fuzzy Systems, Vol.2, pp. 267-278, 1994.

[19]S. Nakkrasae, P. sophatsathit, W.R. Edwards. “Fuzzy subtractive clustering based indexing approach for software components classification”; Vol. 14D, Issue 1, pp. 89-96, 2007.

[20]J. Chen , Z. Qin , J. Jia. “A PSO-based subtractive clustering technique for designing RBF neural networks”; Evolutionary Computation, CEC 2008, (IEEE World Congress on Computational Intelligence), IEEE Congress on, pp. 2047 – 2052, 2008.

[21]Fuzzy logic Matlab toolbox, Mathworks, Matlab R2011a.

[22]M. Eftekhari, S. D. Katebi. “Extracting compact fuzzy rules for nonlinear system modeling using subtractive clustering, GA and unscented filter”; Applied Mathematical Modelling, ScienceDirect, Vol. 32, Issue 12, pp. 2634–2651, 2008.

[23]R. Singh, W. Bailey. “Fuzzy logic applications to multisensory - multitarget correlation”; IEEE transactions on aerospace and electronic system, Vol. 33, Issue 3, pp. 1-18, 1997.

[24]M. Shah, P. D. Mehta. “Classification of vehicles using adaptive neuro fuzzy inference system”; Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference, pp. 1-6, 2014.

[25]A. Pandey, N. Gupta. “Stage Determination of Oral Cancer Using Neuro fuzzy Inference System”; Electrical, Electronics and Computer Science (SCEECS), 2014 IEEE Students' Conference, pp. 1-5, 2014.

[26]A. Al-Hmouz, J. Shen, R. Al-Hmouz, J. Yan. “Modeling and Simulation of an Adaptive Neuro-Fuzzy Inference System (ANFIS) for Mobile Learning”; Learning Technologies, IEEE Trans. on, Vol. 5, Issue. 3, pp. 226 – 237, 2012.

[27]T. Ling, M. Rahmat, A. Husain. “ANFIS modeling and Direct ANFIS Inverse control of an Electro-Hydraulic Actuator system”; Industrial Electronics and Applications (ICIEA), 8th IEEE Conference, pp. 370 – 375, 2013.

[28]A. Maurya, D. Patel. “Vehicle Classification Using Adaptive Neuro-Fuzzy Inference System (ANFIS)”; Springer India, Proceedings of Fourth International Conference on Soft Computing for Problem Solving Advances in Intelligent Systems and Computing, Vol. 336, pp. 137-152, 2015.

[29]J. Jang, C. Sun. “Neuro-Fuzzy Modeling and Control”; Proc. IEEE, Vol. 83, Issue 3, pp. 378-406, 1995.

[30]J. Jang, C. Sun, E. Mizutani. “Neuro-Fuzzy and Soft Computing”; Prentice Hall, 1997.

[31]F. Jurado, M. Ortega, J. Carpio. "Power quality enhancement in fuel cells using genetic algorithms and ANFIS architecture"; IEEE International Symposium on Industrial Electronics, pp. 757-762, 2006.

[32]J. Jang. “Fuzzy modeling using generalized neural networks and Kalman filter algorithm”; Proc. Ninth Nat. Conf. Artificial Intell, PP.762-767. 1991.

[33]H. Hui, F. Song, J. Widjaja, J. Li. "ANFIS-based fingerprint-matching algorithm"; Optical Engineering, Vol. 43, pp. 1814-1819, 2004.

[34]K. Assaleh. "Extraction of Fetal Electrocardiogram Using Adaptive Neuro-Fuzzy Inference System"; IEEE Transactions on Biomedical Engineering, Vol.54, Issue 1, PP. 59-68, 2007.

[35]F. Yeh, Y. Lu, C. Li, M. Wu. "Application of ANFIS for inverse prediction of hole profile in the square hole bore-expanding process"; Journal of Materials Processing Technology, Vol. 173, pp. 136-144, 2006.

[36]F. Bashir, A. Khokhar, D. Schonfeld. “Object trajectory-based activity classification and recognition using Hidden Markov Models”; IEEE Trans. on Image Processing, Vol.16, Issue. 7, pp. 1912-1919, 2007.

[37]I. Sbalzariniy, J. Theriot, P. Koumoutsakos. “Machine learning for biological trajectory classification applications”; Proc. Summer Program, Center for Turbulence Research, pp. 305-316, 2002.

[38]R. Fraile, S. Maybank. “Vehicle trajectory approximation and classification”; Proc. British Machine Vision Conf., pp. 832-840, 1998.

[39]J. Owens, A. Hunter. “Application of the self-organizing map to trajectory classification”; Proc. 3rd IEEE Int'l Workshop on Visual Surveillance, pp. 77-83, 2000.

# SESSION

# SCIENCE, TECHNOLOGY, ENGINEERING AND MATHEMATICS - STEM EDUCATION

## Chair(s)

**TBA**

# Pathway from Learning Multimedia Software to Computer Science Education

**Ching-yu Huang**[1], **Jing-Chiou Liou**[2], **Grace Huang**[3]
[1,2]Kean University, [1]chuang, [2]jliou@kean.edu
[3]University of Pennsylvania, [3]ghua@seas.upenn.edu

**Abstract** – *In this paper, we propose a pathway to encourage high school students pursuing computer science study with 3 steps – from application users to basic programmers. Multimedia software and MS-Excel can be adopted to help high school students to work on video, animation, data analysis for homework or extracurricular projects, which could reduce frustration that one would experience when jumping directly into learning computer programming language. This would consequently increase opportunities in understanding software design. High school computer science curriculum should be designed to encourage more students to smoothly transition from applying the tools to their science and non-science subjects into studying computer science or technology related fields. In addition, statistics show that the rate of switching majors is very high in college STEM programs. Therefore, we also offer some suggestions which would help high school students better understand whether a future in computer science is something they want to pursue.*

**Keywords**: STEM education, computer science, High school curriculum, animation

## 1 Introduction

Computer and Information Technology related jobs [9] include many different kinds of positions – researcher scientists, programmers, network architects, support specialists, system analysts, database administrator, network and system administrator, developers for software and web. In software industry, there are some typical functional groups and roles, such as System Engineer for requirements, developer for coding and tester for testing. In colleges, in addition to the traditional computer science, there are more and more schools separated the computer related fields into different concentrations or tracks - Graphic Design, Game and Interactive Media Design, Human Interaction, Information Technology, Information Science. Many of these computer related jobs and studies do not require strong programming skills. Therefore, it is very important to firstly give an overview of computer fields and interesting hands-on tasks to the students in secondary school to trigger their interests in learning information technology and computers.

Projects and competitions can be an excellent way to encourage students in middle and high schools to engage with information technology; because there are clear objectives for students to achieve. There are many popular competitions for students in middle school and high school to compete in the United States. National History Day (NHD) [1] and Technology Student Association (TSA) [2] are two well-known extracurricular competitions in the United States. In these competitions, students advance from regional, state to national levels. Of NHD's four categories, two of them – websites and documentaries – require students to utilize computers and technology to create their projects. This encourages students to explore the usefulness of computers to accomplishing tasks. TSA's competitions focus more on science and technology. For instance, SciVis, Open Source Software, and Robotics competitions strongly require students to utilize computer technology. For NHD competition, students only use the software and do not need to code in detail. On the contrary, students who participate in TSA may need stronger knowledge and interests in technology and computer science.

In addition to academic competitions, there are also many non-academic opportunities in which students can utilize technology to work on projects, such as the Girl Scout Gold Award project [3], which is the highest honor in Girl Scouts, and the Boy Scout Eagle project [4], which is the requirement for the highest Eagle rank. Both projects, which students often compete during high school, require significant and long-term contributions to the community. Many Eagle Scouts work on physical projects such as constructions, and many Girl Scout Gold Award projects are done in community service. With the advent of modern technology, students are also allowed to have their projects mainly based on computers or other technologies as well. A study [5] showed that the girls of middle school students could be interested in studying computer science, if they were able to exposure to interesting computer games. Here, we will discuss how IT projects can help high school students understand the technology and become interested in studying computer science related fields in college.

Therefore, the first step is to let the beginners experience their interesting software.

## 2 Software to Create Educational Multimedia

Computer fields become very diverse in past decades and it is helpful for students to explorer the fields from software applications, not jumpstart from programming. This way, it will give students as the end-user role to understand in high-level about software features, user interfaces, and computer-human interaction.

As mobile devices gain popularity, more and more educational multimedia materials are being produced. That also encourages more and more students use Window Media, Apple iMove, and Adobe Premiere to make multimedia projects. Students can utilize the software to design theme, add texts, and edit video and audio clips to make multimedia production. Students can present their creations for class work and competitions, and even share them through Youtube, Facebook, and other social media.

Working with these software triggers students' interests in pursuing digital media design, movie-making, or software design fields in colleges. These majors could be under College of Design, College of Arts & Sciences, or College of Engineering. The curricula could differ very much between these colleges, so it is very important for students to understand the differences in education objectives. Here, we use the software Autodesk Maya and Adobe Premiere as examples to discuss how the software could impact students who are learning information technology.


**Figure 1** 3D animation

Maya is a very popular animation software used in industry but is free for academic use. In order to make competitive projects, students have the option to make 3D animations as shown in Figure 1. This requires them to learn and understand how to realistically design, move, and construct 3D-models, realistically light the environment, and understand some basic computer graphics such as transformation – translation and rotation to move objects smoothly. Students can combine the animated objects with sound and text to make an animated movie and save the projects into AVI, MP4, or MOV formats in High Definition (HD) resolution.

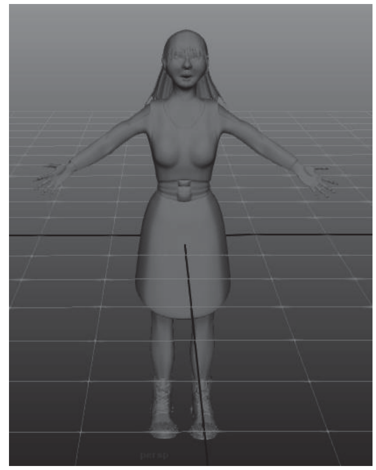


**Figure 2** 3D Modeling

The 3D-modeling process, shown in Figure 2, will allow students to connect to mathematical concepts better, especially in coordination. This will lead students to do better in algebra. Algebra is the core curriculum of mathematics in middle schools and the foundation for all later mathematics. The computer graphics transformations will give students strong training in geometry concepts while they move the objects around in their projects. Geometry is normally taught in $8^{th}$ or $9^{th}$ grade, which is an important stage for students to explore and decide if they want to pursue the STEM areas. Therefore, since Maya is free to academia, schools could offer classes that use the software and encourage students to work on projects using the software.

Adobe Premiere and iMove are two popular software for making videos. They allow users to crop, edit and merge photos, videos, audio, and text. Although it doesn't require much mathematics, the software has many features for students to work and learn visual effects so it gives them experience with Human Computer Interaction (HCI) and Graphic User Interface (GUI) software design. These two areas are getting more attention in computer science education, because it is very important to have well-designed interfaces on mobile devices. Therefore, a new emerging field of art and software design has been established in colleges over the past few decades. Students who study in art colleges need to learn basic software design, and students who study computer science are encouraged to take art and human behavior related courses to design better software interfaces. Many computer languages have Integrated Development Environment (IDE) such as Eclipse and it provides friendly environment and better way to write programs. It does help a lot to reduce frustration of learning a new language for a beginner in computer science.

# 3 Interests vs. Major in Computer Science

Statistics show that about 69% of bachelor's degree candidates in science, technology, engineering and mathematics leave the field before completing a college degree [6]. While students could be interested in software design and digital media design fields, they might be not sure which areas to pursue and confused by the majors during the college applications. US high schools currently offer core curriculum in mathematics, sciences (biology, chemistry, and physics), humanities, and English. Although some comprehensive high schools could offer elective courses in calculus and computer related course, many schools don't offer computer programming courses such as Advanced Placement (AP) computer science. New Jersey state Assembly has passed the bill 2597 [7]: Provides that beginning with the 2014-2015 grade nine class, Advanced Placement computer science course may satisfy a part of either the mathematics or science credits required for high school graduation. This will help students in high schools truly understand what computer science is.

The web 2.0 provides a lot of static online information for computer science education. Teachers and students are able to access free documentation on software or hardware vendors such Oracle, W3School, etc. There are many free websites for people to receive answers about questions they have such as youtube.com, stackoverflow.com, etc. Online learning becomes playing important roles in computer science education because classroom and books are not the only resources for students to get knowledge and answers on all subjects. If schools cannot offer computer programming courses, it is important for the school to advise students to utilize online resource for self-studying to explore computer science.

A course with some basic programming and discrete mathematics knowledge should be included as an elective course for juniors and seniors who want to pursue computer science. The attrition rate was highest for bachelor's degree candidates who declared a major in computer/information sciences [6]. Because there is a gap in the curriculum between high school and college STEM programs, many students have difficulty adjusting the way they study and smoothly transitioning into freshman year. Hence, they might decide that computer science is not for them and they may change to another major that has a more liberal atmosphere that is similar to that in high school. Students might be interested in software development, HCI and GUI design, but these areas are totally different from high school mathematics and science courses they have taken, and most of college computer science courses required to take advanced courses which freshman students might not be good at. This may be a factor for the numerous high school students who have difficulty transitioning during the dramatic changes in their first year in colleges.

In short-term, if high schools don't have the resources to teach computer related topics, they should allow students to take courses at local colleges or online to make sure students truly understand what computer science and engineering are. High schools should also invite computer science and engineering professionals to give seminars for juniors and seniors. In long-term, the education department should design a course in career development for the high school curriculum to let students know the college study, jobs and career for common colleges and fields.

Once the secondary school students are familiar software applications, they could be ready to do some basic programming. Here, we suggest schools to adopt an introductory course that been taught in many community colleges or for non-computer science major students before the students learn any programming languages.

# 4 Computer Science Curriculum for High School

Companies are promoting computer programming to become part of the high school curriculum. However, students could be permanently lost interest if their first computer course is too tough to learn or too theoretical to apply into the real world. It is necessary to have an essential and practical computer science course(s) for high schools that will make learning computer science a more interesting subject. Current US AP-Computer Science curriculum requires a technology teacher capable of teaching computer Java programming, so only comprehensive high schools are able offer full time positions to these teachers. The course is fast-paced so it becomes a little challenging and thus deters many students from taking the course. Data [8] shows that only around 21,139 students in 2011 took AP-Computer Science exam. It is around 0.5% of 4 million high school senior students in the US. If we want to promote computer science in high schools, a better curriculum is needed.

In this section, we will discuss three types of Introduction to Computer Science course: Multimedia and Animation, Data Analysis and Marco Programming using Spreadsheet, and programming in an Integrated Development Environment (IDE). The three types of courses fit to different curriculum design.

## 4.1 Multimedia and Animation

The computer science curriculum in high school can start from utilizing software to create visual effects such as using PowerPoint, Photoshop, iMovie, Adobe-Premiere and data modeling using Maya animation in 9th and 10th grades.

These topics will help students to make multimedia projects and also learn software functions. Shown in Figure 3 is a student animation project that helps others to understand human nerve system.

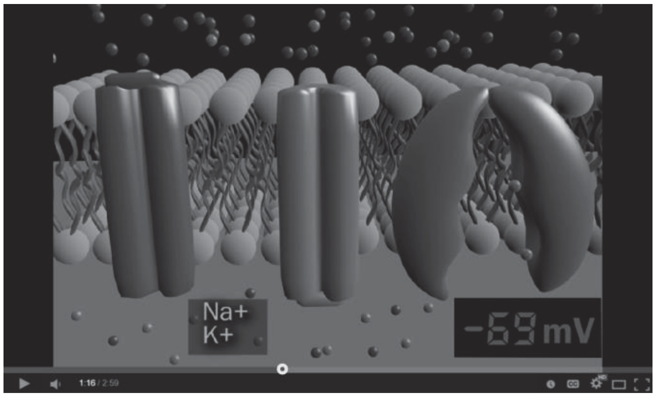



**Figure 3** A student animation project to help understand human nerve system

In this type of course, students are introduced with various software application that are easy to use and the result can be observed immediately. There are no programming skill required. As students can show off their works and gain confidence, motivation to continue on to other computer science courses will be greatly improved.

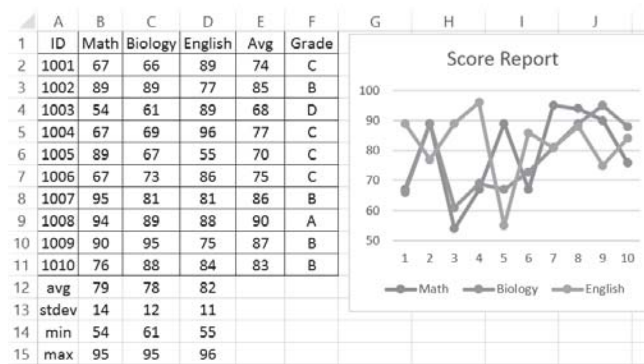
## 4.2 Data Analysis and Marco Programming using Spreadsheet

A more advanced course, comparing with the multimedia in previous section, is to teach data analysis and marco programming concepts. Data analysis and marco programming concepts using MS-Excel, OpenOffice-Calc, Google-Sheets or other software can be taught between 10th and 11th grades.

To simplify the discussions, we will use Microsoft Office to discuss because it is popular among students to write reports in MS-Word, presentations in MS-Power Point, and spreadsheet work in MS-Excel. Teachers and students can utilize the MS-Excel functions to learn basic programming concepts listed below:

- *Variables*: In MS-Excel, cell ID, such as A1 (1st row at column A), B2 (2nd row at column B) can be considered as variables in programming languages.
- *Data type*: integer, double, string used by cell IDs.
- *Assignments*: if A2=1, B2=2, at cell C3, type ***=A1+B1***, it will show 3 in C3.
- *String concatenate*: if A2=1, B2=2, at cell C3, type ***=CONCATENATE(A2,B2)*** , it will show 12 in C3.
- *Boolean – TRUE, FALSE, NOT:* if A2=1, B2=2, at cell C3, type ***=A2=B2***, C3 will show FALSE. If we type =NOT(C3) at cell D3, it will show TRUE at D3.
- *Logic operator*: if A3=FALSE, B3=TRUE, at cell C3 type ***=OR(A4,B4),*** it will show TRUE at C3.
- *IF statement*: if A1=1, B1=2, at C3 type ***=IF(A1=B1,1,0),*** it will show 0 at C3.
- *Complex IF statement*: if A1=1, B1=2, C1=”a”, at C3 type ***=IF(OR(A1=B1,C1="a"),T,F),*** it will show T
- *Aggregate functions – AVG, MAX, MIN, STDEV, COUNT, SUM, etc* : if A1=1, B1=2, C1=3, at C3 type ***=MAX(A1:C1)*** , it will show 3.
- *Substitute*: if A1=”abcd”, at C3 type, =SUBSTITUTE(A1,"ab","AB"), it will show ABcd at C3.

Figure 4 is an example using MS-Excel if-else statements to determine students’ grades of a class.
=IF(E2>=90,"A",IF(E2>=80,"B",IF(E2>=70,"C",IF(E2>=60,"D","F"))))



| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | ID | Math | Biology | English | Avg | Grade |
| 2 | 1001 | 67 | 66 | 89 | 74 | C |
| 3 | 1002 | 89 | 89 | 77 | 85 | B |
| 4 | 1003 | 54 | 61 | 89 | 68 | D |
| 5 | 1004 | 67 | 69 | 96 | 77 | C |
| 6 | 1005 | 89 | 67 | 55 | 70 | C |
| 7 | 1006 | 67 | 73 | 86 | 75 | C |
| 8 | 1007 | 95 | 81 | 81 | 86 | B |
| 9 | 1008 | 94 | 89 | 88 | 90 | A |
| 10 | 1009 | 90 | 95 | 75 | 87 | B |
| 11 | 1010 | 76 | 88 | 84 | 83 | B |
| 12 | avg | 79 | 78 | 82 | | |
| 13 | stdev | 14 | 12 | 11 | | |
| 14 | min | 54 | 61 | 55 | | |
| 15 | max | 95 | 95 | 96 | | |



**Figure 4** Data analysis using Excel cell programming

In order to give students more opportunities to explore basic computer science, teachers could ask students to utilize MS-Excel spreadsheet to do science or math homework, or even analyze their home electric and gas bills using the above basic computer programing components to analyze the data, display them in charts, and submit or present the final reports with charts.

MS-Excel also provides macro for users to write programs in VB to access worksheet and data cells. There are many online Excel macro tutoring [11] and it is very easy for student and non-computer science teachers to learn Excel macro and VB programming. Figure 5 shows the VB codes finding students who have grade >=90 when users click the button on worksheet as shown on Figure 6. Students can use Excel functions and macro to analyze data for school projects and competitions.

```
Private Sub CommandButton1_Click()
    Dim n As Integer
    Dim cell1, cell2 As String
    For n = 2 To 10
        cell1 = "E" + CStr(n)
        cell2 = "A" + CStr(n)
        If Range(cell1).Value >= 90 Then
            MsgBox ("ID" + Range(cell2).Value + " has grade A")
        End If
    Next n
End Sub
```

**Figure 5** MS-Excel macro to find students who has grade A

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | ID | Math | Biology | English | Avg | Grade |
| 2 | 1001 | 67 | 66 | 89 | 74 | C |
| 3 | 1002 | 89 | 89 | 77 | 85 | B |
| 4 | 1003 | 54 | 61 | 89 | 68 | D |
| 5 | 1004 | 67 | 69 | 96 | 77 | C |
| 6 | 1005 | 89 | 67 | 55 | 70 | C |
| 7 | 1006 | 67 | 73 | 86 | 75 | C |
| 8 | 1007 | 95 | 81 | 81 | 86 | B |
| 9 | 1008 | 94 | 89 | 88 | 90 | A |
| 10 | 1009 | 90 | 95 | 75 | 87 | B |
| 11 | 1010 | 76 | 88 | 84 | 83 | B |

Find who has grade A

Microsoft Excel

ID1008 has grade A

OK

**Figure 6** MS-Excel macro display a message box when the program find a student who has grade >=90 after the "Find who has grade A" button is clicked

## 5 Programming in an IDE

Once students become more interested in learning advanced computer knowledge, students can take a dedicated programming course in 11th and 12th grades. This programming course may teach Java, C++ language or any programming language and programming skills using integrated development environment such as Eclipse. This course can also utilize even simpler IDE, such as Alice, that does not incur any complexity from production code structure.

Alice, shown in Figure 7, is designed solely to teach programming theory without the complex semantics of production languages such as C++. Users can place objects from Alice's gallery into the virtual world that they have imagined, and then they can program by dragging and dropping tiles that represent logical structures. Additionally, the user can manipulate Alice's camera and lighting to make further enhancements. Alice can be used for 3D user interfaces.

A controlled study at Ithaca College and Saint Joseph's University looking at students with no prior programming experience taking their first computer science course, the average grade rose from C to B, and retention rose from 47% to 88% [10].

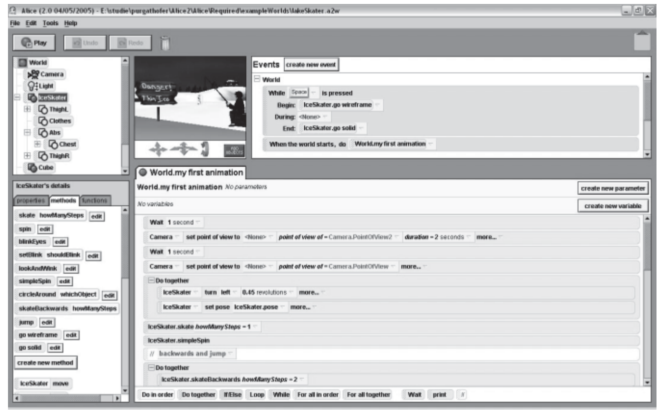


**Figure 7** Alice IDE

## 6 Conclusions

In this paper, we propose a conceptual pathway for students in secondary schools to guide their learning in technology and help them to pursue in computer science fields later. Since today's mobile devices are actually computers and can perform most of he computing tasks, there are more software applications (Apps) developed on the mobile devices than on personal computers. The job market in computer and information technology today has a very strong demand for more skilled people. In the United States, the field is predicted to be 1 million people short by 2020 [12]. It is critical for US to encourage students in middle school and high school to pursue career in the fields of computer science and it's related.

However, it is not necessary to train students just to become software developers, as many computer related jobs do not require programming skills. Those jobs could require students to have good sense of designing user interface, analyzing the data with tools, testing the features, or utilizing information and managing the technology. Therefore, it is more important to keep students interesting in information technology, than learning how to program. The keys to reach this goal are first to encourage students to work on projects using multimedia software and to design a better curriculum utilizing the software, especially in secondary schools. Once students become interested in utilizing computers, there will be more motivation for them to take AP-Computer Science during their high school year, to study computer related studies in university and pursue their career in computer filed later in life.

## 7 References

[1] National History Day (NHD) http://www.nhd.org/

[2] Technology Student Association (TSA) http://www.tsaweb.org/

[3] Girl Scout Gold Award Project, www.girlscouts.org/program/highest_awards/gold_award.asp

[4] Boy Scout Eagle Project, http://www.scouting.org/scoutsource/boyscouts/advancementandawards/eagleworkbookprocedures.aspx

[5] Stewart-Gardiner, C., Carmichael, G., Latham, J., Lozano, N., Greene, J., "*Influencing Middle School Girls to Study Computer Science Through Educational Computer Games*", Journal of Computing Sciences in Colleges, Vol 28, June 2013

[6] Megan Rogers, "STEM-ming the Tide" *Inside Higher ED,* November 27, 2013. https://www.insidehighered.com/news/2013/11/27/study-tracks-attrition-rates-stem-majors

[7] New Jersey Assembly Bill 2597, http://legiscan.com/NJ/bill/A2597/2014

[8] "CS Education Statistics" *Exploring Computer Science*. http://www.exploringcs.org/resources/cs-statistics

[9] "Occupational Outlook Handbook - Computer and Information Technology Occupations", http://www.bls.gov/ooh/computer-and-information-technology/home.htm

[10] M. Moskal, D. Lurie, and S. Cooper, Evaluating the Effectiveness of a New Instructional Approach.

[11] http://www.excel-easy.com/vba/create-a-macro.html#developer-tab

[12] Allie Bidwell, "Tech Companies Work to Combat Computer Science Education Gap", US News, Dec. 27, 2013 http://www.usnews.com/news/articles/2013/12/27/tech-companies-work-to-combat-computer-science-education-gap

# Improving Technical Writing through Professional Expression Exercises in a Computer Science Graduate Program

Meline Kevorkian#1, Gregory Simco#2

[1] *Office of Academic Affairs, Nova Southeastern University*
*3301 College Avenue, Fort Lauderdale, FL 33314, USA*
melinek@nova.edu

[2] *Graduate School of Computer and Information Sciences, Nova Southeastern University*
*3301 College Avenue, Fort Lauderdale, FL 33314, USA*
greg@nova.edu

The 2015 World Congress in Computer Science,
Computer Engineering, and Applied Computing

***Abstract*—Building quality programs based on workforce skills, such as technical writing, is considered an effective means for recruitment and retention in higher education, especially in the areas of science, technology, engineering and math. In computer science, technical writing can be especially challenging due to the complexity of technical documents. This work incorporated the use of professional expression activities in the computer science graduate curriculum at Nova Southeastern University. Additionally, the results support continual building of the quality of the Master of Science in Computer Science and technical writing in the discipline.***

Keywords: Computer Science, Technical Writing, and Professional Expression

## I. Introduction

Writing is the most common form of scientific communication, yet many professors complain that graduate students are too often poor writers. Scientific writing is assumed and not really taught, however, stronger writing skills benefit educational pursuits and science careers. Therefore, the ability to write good scientific articles is a skill necessary to disseminate ideas to the scientific community. Kroll, Schulze & Kern (2014) discuss the importance of reinforcing discourse analysis, which in scientific writing is important because it is taking language beyond the level of a sentence [1]. The value of discourse analysis is that it takes into account context, information extraction, claims and hypotheses, and the identification of methods and protocols.

Professional expression that incorporates technical writing and critical thinking is a necessary skill for STEM careers, particularly in computer science. The literature effectively supports this concept:

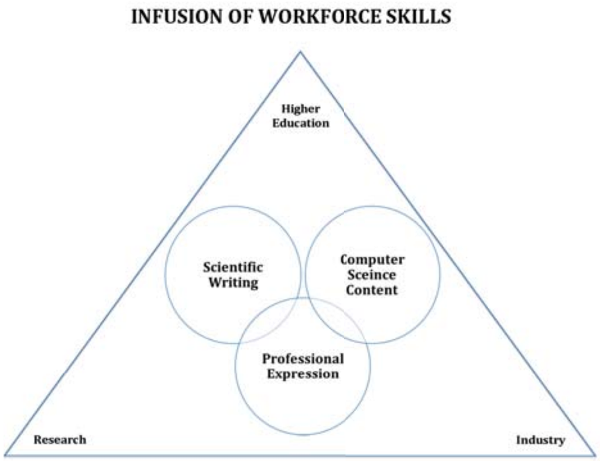
"In the STEM fields, the single most important thing we do to teach our students better communication skills is to teach them how to write a lab report. The reason a lab report is so important is not just because it is the end product of a research project, but because each section of a lab report has particular function that often correlates with different types of communication that a STEM professional needs to use" [2].

Many employers share that new employees often require remediation. According to the survey, *Are They Really Ready to Work? Employers' Perspectives on the Basic Knowledge and Applied Skills of New Entrants to the 21st Century U.S. Workforce*, only 23.9 percent report that new entrants with four-year college degrees have "excellent" basic knowledge and applied skills [3]. Additionally, employers noted new hires exhibited significant deficiencies in written and oral communication skills.

Computer science is a challenging area of study in many respects but the difficulty is often due to poor writing skills, not the concepts in the assignments. This work found that scientific writing infusion into a computer science course improved the comprehension of the concepts as well as the writing. For sustainable success in both industry and research, scientific writing, that we term "professional expression" is equally important to technical knowledge.

The implications for computer science educators are to create graduates who are more agile graduates. Higher education must be connected to research and industry and realize that professional expression, defined as the ability to take technical knowledge and articulate it to a particular audience, is the main ingredient in graduate education. Professional expression combines scientific writing and critical thinking with computer science content. This further prepares future graduates in a workforce where they will move across different jobs and sectors. Workforce agility supported through professional expression in the curriculum provides an added benefit to graduate education in computer science. The diagram below indicates the necessary relationship among higher education, research, and industry to provide a workforce ready curriculum.

Diagram 1




## II. PROFESSIONAL EXPRESSION INFUSION

Working with data collected for the M.S. in Computer Science, the faculty measured the impact of professional expression exercises on improvement in final course products. The standard for measure was the quality of writing (grammar, word selection, clarity, organization, and smoothly flowing discussion), and understanding of the content (accuracy, analysis, and synthesis) related to a chosen focus within the assigned topic area. The intent was to develop a deeper understanding of the content by being able to explain and defend their thoughts and solutions.

The Computer Science faculty evaluated the work through three to four short professional expression exercises. This paper reports on the outcomes and includes a discussion of the rationale for incorporating short feedback intensive exercises early in a course. The findings clearly show the linkages between the improvement in final student products and performances. These professional expression exercises complemented multiple design/coding assignments along with a major research project. The professional expression work was evaluated per individual then discussed as a group in a classroom setting and in asynchronous online discussions. Research supports that teaching experiences have a positive impact on the participants' ability to generate testable hypotheses, create valuable research designs, and write research proposals. [4][5].

The writing assignments were designed to provide early feedback, get writing sample baselines, and build a focus on writing for the content area to better express understanding of the material. The emphasis is on both problem-solving and expression of the solution. Examples of the exercises are found in the table below.

Table 1

| |
|---|
| Example 1<br><br>Answer the following:<br><br>Two computer science (CS) professionals are having a heated debate. CS professional A believes that all systems and applications must provide for an abstraction from the details of the implementation and CS professional B is an advocate of exposing the details. Provide a detailed account (support your claims) of your view. |
| Example 2<br><br>Address the following and be prepared to extend the discussion into a larger group.<br><br>As a developer in a research and development department of a startup company that provides web hosted applications, you are asked to support the advantages and disadvantages of using an open-source operating system in your production environment. Provide a brief and supported explanation of your thoughts. |
| Example 3<br><br>You work for an IT department that supports a company that manufactures, markets, and sells toys. Additionally, the company's website provides web applications including games and shopping. While having a conversation with a fellow IT professional, it was mentioned you should read the following:<br><br>Roche, J. (2013). Adopting DevOps practices in quality assurance. *Communications of the ACM*, *56*(11), 38-43.<br><br>The next day you are asked by your manager to give a brief but informative paper on the value of DevOps in your IT department. |

## III. PROFESSIONAL EXPRESSION RESULTS

The overall results of infusion of professional expression in the M.S. in Computer Science led to an improvement in the written quality of coursework as well as the ability to better articulate solutions and outcomes. In organized design documents, it was noted that students were better able to connect each part of the document to the entire product while showing how each section stands on its own merit. In our experience, the feedback early in the course translated into a decreased repeat of errors from each assignment. Overall there was a decrease in cases of just pointing to literature rather than synthesizing and relating it to the topic at hand. Additionally, the professional expression exercises heightened the learning experience by engaging students in using feedback in a meaningful manner and increased student engagement. The feedback served two important purposes. First, the instructor gains early insight on the student content base and writing ability. Second, the student is provided with feedback to take ownership in their learning and continual improvement. To this end, faculty should be discussing ways to add flexibility to their computer science curriculum to better address the needs of the workforce. By designing a curriculum that promotes critical skills for success we can produce greater researchers and industry leaders in the field of computer science.

## IV. Actions for Improving professional expression

Proficiency in a topic area does not necessarily imply that knowledge can be clearly articulated in written or verbal communication. At the graduate level, just giving an example of what is good and bad is not adequate to support quality work. Professional expression exercises with feedback early on and in short turnaround timelines is necessary to provide a continual improvement model throughout the course. Going through these early agile professional expression experiences provide a clearer understanding of expectations and identification of common errors. Additionally, this iterative approach provided a mechanism for improvement within content areas, reinforcing that it is not just what is presented in a communication but how it is composed. This type of articulation in a scientific communication addresses current professional employment requirements by incorporating real-life skills into the classroom.

## V. Implications of the Findings

There is a lack of literature that provides a conceptual framework for increasing the writing skills for graduate students in computer science and related fields. Additional research on professional expression in the curriculum and preparation for graduate level coursework may lead to practices for graduate-level preparation and may be useful in upper-level undergraduate coursework. Research on scientific writing conducted in the computer science classroom has been limited and not been given enough attention. Thus to improve research and workforce preparation an emphasis on purposeful studies in graduate programs are needed to further emphasize and document effective scientific writing practices across technology disciplines.

## References

[1] Kröll, M., Schulze, G., & Kern, R. (2014). A Study of Scientific Writing: Comparing Theoretical Guidelines with Practical Implementation. *Proceedings of SADAATL*, 48-55.

[2] Bennett Jr., P. (n.d.). Using Rubric to Teach Science Writing. Essays on Teaching Excellence. *Toward the Best in the Academy, 20*(8).

[3] Casner-Lotto, J., & Barrington, L. (2006). Are they really ready to 31 work? Unites States: The Conference Board, Corporate Voices for Working Families, Partnership for 21st Century Skills and Society for 32 Human Resource Management

[4] Feldon, D., Peugh, J., Timmerman, B., Maher, M., Hurst, M., Strickland, D., ... & Stiegelmeyer, C. (2011). *Graduate Students' Teaching Experiences Improve Their Methodological Research Skills. Science, 333*, 1037-1039.

[5] Timmerman, B. E. C., Strickland, D. C., Johnson, R. L., & Payne, J. R. (2011). Development of a 'universal rubric for assessing undergraduates' scientific reasoning skills using scientific writing. *Assessment & Evaluation in Higher Education*, 36(5), 509-547.

# Bridges to STEM Careers: A Collaborative Approach[1]

***Sadegh Davari, Sharon Perkins-Hall, Krishani Abeysekera***

*Division of Computing and Math, University of Houston-Clear Lake, Houston, Texas, USA*

***Jim Meeks, Nathanial Wiggins, Norm Liebling***

*San Jacinto College District, Pasadena, Texas, USA*

***Abstract*** *- The Bridges to STEM Careers (BSC) project is a collaborative effort between the University of Houston - Clear Lake (UHCL) and three campuses of the San Jacinto Community College district (SJC) to increase attainment of STEM associate and baccalaureate degrees, as well as to facilitate and increase transfer of STEM students across the institutions. The project focuses on increasing student engagement and persistence through the development of face-to-face and technology-enhanced learning networks, both within and across campuses. Supporting initiatives include extended orientation programs, peer and faculty mentoring, career-awareness and enhancement activities, and modest financial aid. The intellectual merit of the project lies in the strong partnership between the baccalaureate and associate institutions, well-designed and coherent project elements built on literature-based best practices and successful pilot studies, clear implementation strategies, well-planned evaluation, and extensive involvement of local STEM industries. In this paper we will provide a brief overview of the project elements, hands-on student activities and the lessons learned so far.*

**Key Words:** retention, bridge program, peer mentoring, hands-on STEM activities, academia-industry collaboration.

## 1 Introduction

The Bridges to STEM Careers (BSC) is a five-year project funded by the NSF STEP program. BSC project is unique among other STEP projects in the sense that it is a collaborative effort among a university and a community college system (SJC campuses). Our goal is to identify students with STEM talent, get them excited, and then guide them to succeed in a STEM field. Recruited students will be mentored by their peers, by faculty, and by industry partners. The overall goal of the project is to increase the number of well-educated STEM graduates for the benefit of local and global economy. The project meets the needs of our institutions to engage, recruit, retain and graduate students in STEM fields. The primary obstacles our students face in degree attainment are lack of engagement (with peers, faculty, campus services and career opportunities) [1, 2] and financial needs. These obstacles are addressed through team-building, project-based activities that involve students from all four campuses and facilitated by mentoring, extended orientations and a STEM Cyber-Center website built with content from participating students. Technology is the theme for our program, and continuous exposure to STEM career opportunities in the Houston area is interwoven into our activities. We have been focusing on the following STEM Fields: Computer Science, Computer Engineering, Computer Information Systems, Information Technology, Mathematics and Physics.

Our project elements are based on literature based best practices [3] and successful pilot studies [4, 5]. Student participants of the project will be given many opportunities to excel in their field of study. They will have opportunities to get directly involved in the design and the development of the project's Cyber Center. They will have opportunities to practice team

[1] This work is supported in part by the National Science Foundation award: DUE-1317490

work, to get internships, and to get exposed to many emerging technologies such as robotics, mobile app development, computer forensics, cyber security, computer game programming, web design, and video editing.

# 2 Project Activities

In working toward getting more STEM students to complete 2-year and 4-year degrees, the BSC Executive team works closely and meets regularly each month. A main purpose for student activities is to make students from other campuses feel welcomed, particularly those from UHCL's partner community colleges, so that they might be more inclined to attend UHCL and complete a bachelor's degree. In the first two years of the NSF STEP grant, various activities have involved the BSC Executive team, community college students, UHCL students, an external evaluator, and advisory board members.

## *2.1 BSC Club*

A BSC club is active at all four campuses. The clubs use tutoring, mentoring, learning centers, programs and courses specifically developed for at-risk and first-year students, academic advising and career awareness. The clubs at all four campuses are considered interchangeable, and students from any of the campuses are welcomed to participate at all campuses. There are currently 43 students registered on the website. Students are invited to attend BSC club events, as well as to participate in Tech Fridays and the Annual Programming Challenge.

## *2.2 Peer Mentors*

Each campus has a team of students who have various responsibilities, including web design, development, and management; tutoring; running the BSC club activities; and preparing and administering the annual programming challenge. Student mentors at all campuses attend targeted classes and serve as in-class mentors. The mentors work to plan an all-campus event called Tech Friday, which involves a hands-on learning activity that is usually held at UHCL. The student mentors also hold regular office hours and act as tutors.

## *2.3 STEP Meetings*

Four members of the BSC Executive team participated in the 2014 STEP Annual Meeting, March 5-7 in Washington D.C. There were several opportunities to meet with experienced faculty to learn about ways to retain students in STEM disciplines. The team prepared a poster that was shared with other grant recipients [6]. The project team members stay connected with other STEM activities through the announcements and notification from STEM Central. One team member participated in a meeting of Texas STEP grantees in College Station, Texas, in May 2014. In May 2015, another team member participated in a STEP grantees meeting in Orlando, Florida. The purpose of this meeting was to meet with members of a local workforce board and learn possible ways of collaboration for the purpose of sustainability.

## *2.4 Tech Fridays*

Tech Fridays are designed by faculty, students, and club members to create collective learning opportunities. Industry and academic speakers are invited to provide relevance.

In the first semester of the grant, the team hosted a Tech Friday on robotics. Students worked with Arduino boards and learned how to program the boards and cause LEDs to light. Fifty students attended this event.

Two Tech Friday events were held in the second semester. The first was a hands-on computer forensic investigation in which students worked with a simulated case that required use of forensic software tools [7]. A Houston police officer talked with students about his experience as a computer forensics specialist. Participants were shown a video of a staged crime scene in which a student's laptop was found, containing a stolen exam key. The laptop was entered into evidence, and participants were given the task to determine the type of crime and the evidence they would extract from it. The second Tech Friday gave students experience with using a Raspberry Pi.

Two Tech Friday events were held in the third semester. The first one was on Web Development. An experienced professional was invited to give background and a lecture on careers in web development. Students with no experience in programming were instructed to produce a webpage. Forty students registered and attended this event. The second event gave students an introduction to video editing. Students were instructed on editing sound, image, and video using freely available software. Seventeen students registered for this event.

## 2.5 Programming Challenge

### 2.5.1 Robotic Programming Challenge 2014

Teams were invited from surrounding community colleges, high schools and UHCL for a programming challenge. The challenge required teams to develop programs in three areas: 1) Morse code to blink specified letter patterns, 2) use of sensors, such as temperature and water, and 3) moving a robotic tank through an obstacle course. The events were timed and checked for completion.

The team that won the competition was offered summer internships at a local industry. Scholarships to UHCL and prizes were given to winners in two categories: beginner and advanced. Teams were self-determined as to their expertise. Prizes were a very well-received motivator.

Credexo, a local company owned by a UHCL alumnus, sponsored the event. Additional support was provided by AtLink Inc., Barrios Technology, and various other companies in the community. Faculty members from community colleges, high schools and UHCL participated actively and served as judges, technical program supervisors and masters of ceremony. Industry representatives from Credexo, Barrios Technology and SAIC also served as judges.

The challenge consisted of two levels of competition: beginners and advanced robotics programmers. The event was limited to 15 teams, with a maximum of 4 members per team. Eight beginner teams and seven advanced teams participated in the event.

During the challenge, teams spent 50 minutes at one robotics station, attempting to complete up to 4 different tasks, before rotating to the second and third station. Each task was assigned a point value, and each task had to be completed and demonstrated before receiving instructions for the next task.

In the Beginner category, each member of the winning team went home with $400 scholarships to UHCL and a robotic kit priced at $100. The second place team members each received $250 scholarships to UHCL and a Google Chromecast priced at $35.

In the Advanced category, the winning team members each received a $500 scholarship to UHCL and a tank robotics kit priced at $140. The second place team members each received $350 scholarships to UHCL and a Google Chromecast.

The overall winners of the competition received an opportunity to interview for two summer internship positions at AtLink Incorporated. After the interview, AtLink offered all 4 students summer internships.

### 2.5.2 STEM Challenge 2015

STEM Challenge 2015 was sponsored by Credexo Inc., Esyntaxis and AtLink. Additional support was provided by various other companies and restaurants in the community. Mentors helped with challenge problem definitions, setup, and facilitation. As in the previous challenge, faculty members and industry representatives served as judges.

The challenge was open to all high school and undergraduate students. Eight different high schools and 6 different community colleges and universities were represented in the challenge. The challenge consisted of two levels of competition, with each team rotating through three stations: Game of Clues to challenge their math knowledge, Creative Inventors to showcase their imaginative skills and Robot Adventure to test their problem solving and programming skills. The event was limited to 18 teams, with a maximum of 4 members per team. 12 beginner teams and 6 advanced teams participated in the event.

During the challenge, each team spent 50 minutes at one of the three stations, trying to complete tasks that were progressively more challenging, before rotating to the second and third station. Each task was assigned a point value, and each task had to be completed and demonstrated before receiving instructions for the next.

In the Beginner category, each member of the first place team went home with $400 scholarships to UHCL along with an opportunity to interview for 1 of 4 summer internships at AtLink. Each member of the second place team in this category received a $250 scholarship to UHCL along with a complete Arduino turtlebot robotics kit.

In the Advanced category, each member of the first place team received a $500 scholarship to UHCL along with an opportunity to interview for 1 of 4 summer internships at AtLink. Each member of the second place team in the advanced category received $350 scholarships to UHCL, along with a Logitech Bluetooth Keyboard.

Three teams were awarded Station Master Status, and these teams went home with Bluetooth speakers, Amazon Fire TV sticks or Motorola Portable Battery Chargers. The first and second place winners of both categories were not eligible for this prize.

Thanks to several departments at UHCL and area restaurants and vendors, participants won many door prizes, such as 1TB solid state external hard drives, flash drives and gift cards.

### 2.6 *Extended Orientation*

A 3-day extended orientation in August 2014 brought together many students majoring in STEM disciplines. The community college partners offered participants $100 towards textbooks for attending all 3 days of the summer orientation, which began with the first day at each of 3 community colleges. The second day was held on one of the partner campuses, and included a motivational speaker. The third day was held at UHCL. The third day activities consisted of a scavenger hunt that required students to find specific offices, labs, and landmarks on the UHCL campus. Teams had to verify that they accomplished certain goals in a particular order by taking pictures with their cell phones. There was also a question and answer session with academic advisors and faculty of different STEM areas that students found particularly helpful.

Each day was a morning event that ended with lunch at noon. All students receive prizes and giveaways, including a BSC backpack full of school supplies. Of the students that attended, 84% said that they feel more confident about their transition to college as a result of attending and 100% said they would recommend the orientation to a fellow student. Positive comments were offered regarding activities and prizes, and suggestions were given for future orientations. The orientation marked the opening of the BSC clubs for the school year. With more than 30 official members across the 4 campuses each campus was able to maintain officers and budget for each branch of the club. Of the students that attended the orientation, more than 50% stated an intention to continue participating in the club regularly throughout the semester.

## 3 Cyber Center Website

The current functionalities of the Cyber Center include: introduction to the BSC project, a news and comments system, site membership, events schedule and registration, pictures from past STEM events, and archives of past STEM events and resources. The UHCL campus hired a research assistant with the primary responsibility for developing and maintaining the Cyber Center (http://bsc.uhcl.edu/). The planned additional functionalities include: increased interactivity, more useful resources, and social media integration.

## 4 Internal/External Advisory Board Meetings

The internal advisory board members of BSC project consist of UHCL and community college administrators and senior faculty members. The External advisory board members consist of members of neighboring universities and local STEM agencies and industries. We have been meeting with board members annually to report progress and to receive feedback and advice. This gives the BSC Executive team the opportunity to learn of ways to improve the efficiency of our project activities. Some of the actions taken as a result of recommendations made in 2014 by IAB/EAB members include: changing the theme of annual challenge from programming challenge to STEM Challenge, incorporating more hands-on activities, and involving outside experts in project activities.

## 5 External Evaluator

BSC project has an external evaluator who evaluates the effectiveness of major project activities. The evaluator collects a mix of qualitative and quantitative data. The qualitative data is collected via face-to-face interviews with students, open-ended survey questions, and observations during the events. The quantitative data is collected via pre and post surveys that measure students' preparedness, self-perceived aptitude, and attitudes/interest pertaining to STEM. Pre and post surveys are generated via Survey Monkey. A link to the survey is provided to each student and students may access the link using cell phones or laptops. Students also could opt to take a hard copy of the survey. The evaluator stays in compliance with all UHCL IRB restrictions. Survey data is analyzed using SPSS. The results are provided to the project executive team for possible actions and it is reported to NSF in our annual reports. Survey results are available by request.

# 6 Conclusions and Lessons Learned

One primary subject of difficulty to undergraduate students is mathematics, and in particular, college algebra. To combat the extremely low passing rates, a more beneficial structure was implemented using Marzano Taxonomy, and community college mentors attended class where appropriate. Additionally, mentors held tutoring hours outside of class in the Math and Engineering Labs. The study showed that the number of students with passing grades of A through C, compared to the number of students present on the initial official roster of the course is exactly 50% in the section with mentoring support vs 44% in the other section.

Mentoring along the critical pathway of algebra, pre-calculus, and Calculus I, along with introductory computer courses allows students to be ready for their major courses. The last year at the community college level and the transition to a 4-year university is deemed equally important to ensure a smooth transition. Thus, the inclusion of a mentor into the final year of studies allows for students to become engaged in university-style preparation for courses such as Calculus III, Differential Equations, and Programming for Engineers. In response to suggestions by the external advisory board for creating industry-driven group projects, advanced interdisciplinary projects have been implemented and supported by the BSC peer network. Some recent accomplishments are listed below [8-10]:

- First place in the Bayer Alka Rockets Challenge at the George R. Brown Convention Center during the 100 Year Starship Symposium.
- Two entries into the Texas Space Grant Consortium Design Challenge for a Mars Drill and Automation and a Smartwatch Design.
  - Second top team overall
  - Second best poster
  - Second best model
  - Third best oral presentation
  - Best Team Logo

In addition to activities and mentoring, the grant supplies scholarships. SJC gives 60 scholarships of $500 each year to students across the district. All scholarships have been awarded thus far. The financial support in conjunction with formal and informal activities is intended to provide a seamless education. Further activities of the grant include a Programming Challenge with numerous sponsors and prizes and a Cyber Center which hosts the sites for the events.

**Some observations:**

1. Students will make time to attend if there is an opportunity to learn about STEM topics.
2. Providing opportunities for students to learn about STEM topics provides an opportunity to get students excited about STEM careers.
3. Having Tech Fridays provides students an opportunity to build relationships with other students interested in STEM careers.
4. The students have a chance to meet faculty and students from other campuses and UHCL at Tech Fridays.
5. Tech Friday exceeded my expectations: 50% of the students responded strongly agree, 37.5% agree.
6. I would like to attend another Tech Friday: 79.17% of the students responded strongly agree, 20.83% agree.
7. When asked what was most liked about Tech Friday, here are some responses:

   "The hands on working"

   "That I got deep into how the computers actually work to access information. I learned tons of things I never knew about. I am definitely coming back next time."

   "This was an enjoyable experience. The infusion of mathematics was nice. Students need to know number theory!"

   "I liked the relaxed, fun atmosphere."

8. When asked what they would change about Tech Friday, here were some responses:

   "At this point, nothing. Having workshops on different technology subjects is working well for me"

   "The cheesy video and scene was awkward. More so the scene before the video than the video itself"

   "Colder room"

"Big turnout – needed additional stations but fun time! Would like to see more robotics demonstrations."

9. Programming Challenges give students a chance to use their knowledge to complete projects in STEM topics. The Programming Challenge and the STEM Challenge made our students want to learn more about computer hardware, robotics and programming.

# 7 References


1. Jolly, Campbell, and Perlman, "Engagement, Capacity and Continuity: A Trilogy for Student Success" (GE Foundation, September 2004)

2. Chun-Mei Zhao and George D. Kuh, "ADDING VALUE: Learning Communities and Student Engagement", Research in Higher Education, vol. 47, 2006, pp 89-109

3. American Society for Engineering report, "Going the Distance: Best Practices and Strategies for Retaining Engineering, Engineering Technology and Computing Students" (Washington, D.C., 2012)

4. Davari, S., Abeysekera, K. and Yue, K., "Building STEM Awareness through Programming Competition", the Fourth Annual Texas Engineering and Technology Consortium Best Practices Conference, Austin, February 11-12, 2009

5. Abeysekera, K., Davari, S., Yue, K., Brown, E., Kent, M., Betts, P., & Meeks, J., Success through Academic Recognition (STAR): Sustaining and Expanding UHCL and SJC TWD Computer Science Scholar Program, the third annual Texas Engineering and Technology Consortium Best Practices Conference, Dallas, Feb. 28, 2008, pp 7-9. www.thecb.state.tx.us/index.cfm?objectid=8828378A-D358-8867-5E14BDC65C9860B9

6. Davari, S., Perkins-Hall, S., Abeysekera, K, Reilly, A., Wiggins, N., Meeks, J., Liebling, N. "Building Bridges Between Community Colleges, a Four Year University, and STEM Careers", Poster presentation at 2014 Annual NSF STEP Grantees Conference, Washington, D.C., March 6-7, 2014.

7. Abeysekera, K., Zhang, T., Perkins-Hall, S., Davari, S., "Creating awareness of the field of cyber forensics with a simulated digital crime scene investigation", 19th Annual Colloquium for Information Security Education (CISSE), June 14, 2015, Las Vegas, NV.

8. Wiggins, Perkins-Hall, et. al, "First Year Engineering Experience, Resource Management in Completion of an Arduino Engineering Project and Its Industrial Applications", First Year Education Experience Conference, College Station, TX , August 7-8, 2014.

9. Wiggins, Yue, et. al, "Microgravity Science and Technology, Acoustic Flame Suppression Mechanics in a Microgravity Environment", Microgravity Science and Technology, March 21, 2015.

10. Wiggins. N, Mota, J.D., "STEM-Bot Mentors", Denver, CO., November 12, 2014.

# Early College Model Approach: Establishing Best Practices to Increase the Number of Underserved High School Students in STEM

**Dr. Sherry Kinslow [1], Dr. Lethia Jackson [2], Dr. Christine Barrow [3] and Dr. Mara Doss[4]**
[1]Psychological and Sociological Sciences Department, Prince George's Community College, 301 Largo Rd, Largo, MD, USA
[2] Department of Computer Science, Bowie State University, 14000 Jericho Park Rd, Bowie MD, USA
[3]Dean of STEM, Prince George's Community College, 301 Largo Rd, Largo, MD, USA
[4]Sr Academic Administrator to the. VP for Academic Affairs, Prince George's Community College, 301 Largo Rd, Largo, MD, USA

**Abstract** - *Prince George's Community College (PGCC) in partnership with Prince George County Public Schools (PGCPS) opened the first middle college high school in Maryland designed to provide underserved high school students with the opportunity to earn college credit while working toward their high school diplomas. This early college model provides an opportunity for 9-12th grade students to simultaneously complete high school and up to two years of college tuition free by including experiential learning outcomes. PGCC, along with PGCPS, bring this experience to the Information Technology Early College (ITEC) Program. The overall vision of the ITEC program focuses on creating a standard approach that provides Maryland's workforce skilled individuals for both the professional and "blue collar" STEM economy. The stakeholders are the students, parents, PGCC faculty, and high school personnel. This paper presents best practices of the ITEC program for all stakeholders.*

***Keywords:*** *High School, College Credits, Earn college credits early, Middle College, Early College*

## 1 Introduction

Prince George's Community College (PGCC) is one of only eleven community colleges nationwide to be named a National Center of Academic Excellence in Two-Year Education in Information Assurance as designated by the National Security Agency and the Department of Homeland Security. This designation indicates that PGCC is a recognized leader in IT education. PGCC has also been an innovator in the early college movement. PGCC, in collaboration with the Prince George's County Public Schools (PGCPS), opened the Academy of Health Sciences (AHS) at Prince George's Community College, which is the first middle college high school in the state of Maryland. This innovative middle college high school provides underserved (e.g., minority, economically disadvantaged, etc.) high school students in Prince George's County with the opportunity to earn an associate degree tuition free while working toward their high school diplomas [2]. The AHS model includes experiential learning to increase diploma/degree completion and learning outcomes. PGCC's successful experience with AHS provides it with the ability to recognize and address the ongoing national issue of how to increase the number and diversity of students in the education/career pipeline who pursue degrees/careers in high growth, high need areas in science, technology, engineering and mathematics (STEM). PGCC, along with PGCPS, bring this wealth of experience in the early college/middle college model to the Information Technology Early College (ITEC) Program.

The ITEC Program involves PGCC, PGCPS, and the Chesapeake Math and IT Academy (CMIT), a public charter school that is part of the PGCPS system, in collaboration to form the foundation for an early college program. The focus of CMIT, which opened in August of 2011, is in science, technology, engineering, and mathematics (STEM) disciplines. The ITEC Program's first cohort, eighth grade students rising to the ninth grade, established baseline data and fine-tuned the development and implementation of the infrastructure, curriculum and professional development, needed to represent a new paradigm for increasing high school students in STEM specifically Information Technology fields [2].

## 2 Background

The overall vision of the ITEC program is to educate, credential, graduate, and employ students with knowledge and skills that serves as a pipeline to Maryland's workforce for both the professional and "blue collar" STEM economy of the 21st century and beyond. The mission of the ITEC Program is to transform students' lives by providing them with innovative, rigorous learning, training, and career experiences that meet or exceed industry standards. The goals to achieve the overall mission of the ITEC program are:

*Goal 1:* To establish a seamless pipeline of students from high school to community college to pathways to four-year colleges/universities and/or employment in STEM or STEM related careers, industries, and fields [1].

*Goal 2:* To establish the personnel, academic resources, and technology infrastructure to support the academic success for ITEC cohorts.

*Goal 3:* To provide accessible and effective academic and career counseling services that enhances student retention, persistence, and completion of high school and matriculation into postsecondary education or entry into the workforce in IT/STEM occupation /fields.

Goal 4: To establish partnerships with industry to design strategies to develop a model pipeline approach that will address recruitment and retention issues of ITEC students as they make their way through the educational and career pathways for the IT industry.

# 3 ITEC Program Framework

There are several strong points of the ITEC Program one of which includes the fact that this program is attracting students into the STEM disciplines. Enrolling more students in STEM fields is important to higher education in general given the relative dearth of US students entering these disciplines in comparison to students in other countries, a trend which some feel could seriously impact the global economic standing of the US in the future [7]**.** However, not only are students being drawn to STEM, students who are women and minorities dominate the participants in the ITEC program, another particularly strong point of the program since such individuals are typically grossly underrepresented in the STEM fields [3]. Upon completion of the ITEC program, moreover, participating students will receive a two year degree at the same time as their high school diploma and could feasibly enter the IT workforce even as early as after high school if so desired.

Additionally, the ITEC program features innovative services for its student participants. As part of the ITEC program, students are provided with laptops to use to assist in completing their assignments and to facilitate their access to a Virtual Learning Lab. Students use wireless access points to support their access to the lab. This process of virtualization thus allows the students to log in at CMIT and then be able to access college applications remotely. Through virtualization students have the ability to access state of the art technologies that can be used to enhance learning, empower an active learning environment as well as gives the students early experience in using the tools for the IT discipline.

Another plus for the ITEC program was the early creation of an Advisory Committee that was charged with overseeing the progress of the program in addition to its implementation and sustainability. The Advisory Committee is designated as the agent to assist with: (1) Developing and implementing academic and professional procedures, (2) Establishing performance standards, (3) Developing and implementing budgets and financial procedures, (4) Providing supervision for the development of an annual report and other essential reporting documents, and (5) Approving the annual calendar for dually enrolled students. Members of the Advisory committee were drawn from PGCC, CMIT, the PGCPS system, other institutions of higher education, and the IT industry itself. The ITEC Advisory Committee meets monthly and has designated work groups for: curriculum, professional development, a summer bridge program, and technology.

# 4 Data Collection

The first year of the ITEC program (2013-2014) an evaluation was conducted by two educators with prior experience in grant writing, project management, and evaluation. The evaluation focused on assessing the success of the ITEC program by surveying its primary stakeholders. The stakeholders identified included: the student participants, the parents of the participating students, the PGCC faculty who taught the college courses, relevant personnel at CMIT including the newly appointed Program Coordinator, and the high school Teachers of Record (TOC) for the college courses taught at CMIT (in addition to a PGCC instructor, each ITEC course must have a TOC provided by the county school system).

While some quantitative data was collected, the majority of the data collection instruments provided qualitative data. The qualitative data was collected from questionnaires, interviews, and focus groups. All stakeholders participated in the data collection process. Three of the four stakeholder groups (the students, parents, and CMIT personnel) met with the evaluators in separate sessions. Each stakeholder group was administered questionnaires that required written responses. Following the completion of the questionnaires, the evaluators conducted a focus group with each of these constituency groups. The fourth stakeholder group, the PGCC faculty members, was interviewed by phone by one of the evaluators, using a questionnaire designed for this purpose by the reviewers.

Based on the findings of the data collection, the evaluators developed a compilation of "best practices" associated with each of the stakeholder groups. These "best practices" articulate in greater detail the activities and issues which need to be addressed for each constituency group.

## 4.1 Student Participants

CMIT's first cohort, eighth grade students rising to the ninth grade, established the baseline data for the program. Student selection to participate in the ITEC program was based on grade point average, Accuplacer placement test scores, recommendations provided by the school counselor and teachers along with parent statements and interviews.

Table 1 describes the criteria and evidence required as part of the selection process.

Seventeen of the 30 recruited students proved eligible to participate in the ITEC program according to the selection criteria. This number was later reduced by three, so that 14 students constituted the first cohort. These students enrolled in courses taught at CMIT by PGCC faculty, taking Introduction to Information Technology (INT 1010) in fall of 2013 and Introduction to Speech Communication (SPH 1010**)** in spring of 2014. As part of the activities associated with ITEC for the 2013-2014 academic year, the students took a field trip to Lockheed-Martin Corporation. The Lockheed-Martin Corporation are one of PGCC partners. This experience was assessed by the evaluators using a questionnaire administered to the students by the Program Coordinator at CMIT. Most students indicated that they enjoyed the field trip or at least liked some things about it, and all who participated in the survey would recommend it to a friend.

The evaluators met with the students at CMIT to discuss their more global experience with the ITEC program. According to the results of the student focus group and written questionnaire, most students additionally expressed concerns that they did not feel that they were adequately prepared for the challenge of college work and felt they could have performed better in their classes with more advanced preparation. They also wanted more college teachers who use active learning pedagogy [6]. (See Table 1)

## 4.2 Parents of Student Participants

The evaluators met with the students' parents at CMIT prior to an orientation meeting. According to the written questionnaire and the focus group results, the parents were appreciative of their child's being able to complete an associate's degree at no cost to them while the child was still enrolled in high school. The parents' concerns primarily focused on the college curriculum and its interface with the high school curriculum. Many of the parents were not aware of the differences in the culture of secondary school versus the culture of college. (See Table 1)

## 4.3 PGCC Professors

Two faculty members from PGCC taught courses at CMIT during this inaugural year of the ITEC program. Both PGCC faculty members participating in the ITEC had prior high school teaching experience. According to the telephone interview results, the PGCC faculty had some concerns. The first semester that ITEC began, there was no program coordinator specifically assigned at CMIT to serve as a liaison between the High School and the College, a situation that proved to be problematic. However, CMIT officially designated such an individual for the spring semester. The PGCC faculty had additional concerns about the need to further differentiate for students and for parents the culture of high school from that of college. The maturity level of some of the students to be able to handle college work was questioned. More professional training for college faculty in teaching the high school students was advocated [5]. (See Table 1)

<table>
<tr><th colspan="2">Best Practices for Students,<br>Parents and PGCC Professors</th></tr>
<tr><td>The Students should be:</td><td>• Given an orientation to the ITEC program, college culture, and expectations for college students<br>• Screened for academic ability (Accuplacer), but also for interest (i.e. Holland scale, etc.)<br>• Provided a schedule of courses for the duration of the ITEC Program<br>• Advised on courses that meet High School as well as ITEC Program's requirements for graduation<br>• Provided with general education course options whenever possible<br>• Provided a midterm course grade<br>• Given an assessment/test that helps the teacher identify the concepts /topics that students are struggling with which may cause a low grade in the class<br>• Required to attend study halls to enhance course performance<br>• Required to attend a winter/summer session as an introduction to spring/fall courses (Dec-Jan/May-June)<br>• Required to attend all field trips, presentations by guest speakers, seminars, workshops, and conferences as part of exposure to ITEC careers<br>• Given a satisfaction survey each semester/year</td></tr>
<tr><td>The Parents should be:</td><td>• Oriented to ITEC Program's course schedule<br>• Oriented to the High School and ITEC Program's requirements<br>• Given the schedule of courses for the program<br>• Oriented to the college culture and expectations<br>• Involved in the semester meetings with CMIT coordinator and PGCC personnel for ongoing communication<br>• Given a satisfaction survey each semester/year</td></tr>
<tr><td>The PGCC Professors should be:</td><td>• Encouraged to provide a midterm grade for students<br>• Recruited and selected based on Professional Development participation and use of active learning techniques<br>• Offered additional incentives such as reassigned time<br>• Involved as a participant in Professional Development which includes active learning strategies, pedagogy for high school students, and the culture at CMIT<br>• Introduced to CMIT personnel prior to teaching at CMIT as part of integration to the environment<br>• Encouraged to build a strong working relationship with the CMIT ITEC coordinator and CMIT TOR<br>• Encouraged to administer formative and summative evaluations</td></tr>
</table>

Table 1:Best Practices for Students, Parents and PGCC Professors

### 4.4 CMIT Personnel

CMIT personnel interviewed include the newly appointed CMIT Program Coordinator and the high school Teachers of Record (TOR) for the college courses taught at CMIT. The responses of the CMIT personnel to a survey administered to them by the evaluators following the session with the students indicate a desire on their part to become more integrated and active in the ITEC program. The verbal interchanges with CMIT personnel reinforced this trend. In particular, the need for the Program Coordinator, TORs, and PGCC instructors to meet in advance of the teaching semester and to preferably all attend professional development activities together was stressed. Developing a solid working relationship between the PGCC professors and the TORs was also highlighted in the focus group discussion. (See Table 2)

### 4.5 PGCC Program Coordinator

Based on the data collected from the constituency groups, the evaluators recommended the creation of a new position as part of best practices. This position serves as a liaison at PGCC. The scope of the position would serve as an interface at PGCC and a coordinator/collaborator with the Program Coordinator at CMIT. The evaluators created a table of best practices for the recommended position of PGCC Program Coordinator. (See Table 2)

### 4.6 Administrators for PGCC

The entire ITEC Program falls under the purview of the overall grant administrators at PGCC who not only wrote the grant proposal, but spear-headed the ITEC Program's implementation, as well as initiated the Program's evaluation process itself. The evaluators worked with these professionals to create a table of best practices reflective of the overarching administrative component of the ITEC program. (See Table 3)

## 5 Summary

Based on the input provided by the surveys, focus groups, and interviews, the evaluators created Tables of Best Practices that articulate the recommendations for the activities and issues that should be addressed with each of the ITEC grant stakeholders including students, parents, PGCC Professors, CMIT Personnel including the CMIT Coordinator and Teachers of Record. Recommendations of best practices were also noted for the PGCC Administrators. The evaluators advocate maintaining the aspects of best practices already being met by the ITEC Program and initiating those not currently being employed.

| Best Practices for CMIT Program Coordinator, TOR and PGCC Program Coordinator | |
|---|---|
| **The CMIT Program Coordinator should:** | • Attend Advisory Board meetings<br>• Interface with PGCC staff and CMIT counselor to prepare schedule of college courses in the ITEC program<br>• Participate in the selection of PGCC professors<br>• Take the lead in Professional Development activities to familiarize PGCC professors with culture at CMIT<br>• Introduce and integrate PGCC professors to the CMIT environment including introduction to CMIT personnel prior to start of each semester<br>• Work with PGCC staff to coordinate CMIT academic calendar of events<br>• Work with PGCC staff to conduct evaluations/surveys |
| **The TOR should:** | • Be introduced to PGCC professors prior to the semester<br>• Cultivate a strong working relationship with PGCC professors<br>• Be responsible for implementing a structured study hall/tutoring session that extends classroom learning experiences for the success of the students |
| **The PGCC Program Coordinator should:** | • Arrange orientation to ITEC for parents<br>• Arrange orientation to college culture for parents<br>• Attend semester meetings with parents<br>• Recruit PGCC faculty<br>• Arrange Professional Development for PGCC staff<br>• Work with CMIT coordinator to establish relationship with coordinator and TOR as well as integrate PGCC staff into CMIT environment and culture<br>• Interface closely with CMIT coordinator to prepare course offerings, schedule, calendar, and technology usage issues<br>• Work with CMIT Coordinators to assist in preparing Professional Development<br>• Work with CMIT coordinator to conduct evaluations/surveys |

Table 2: Best Practices for CMIT Personnel

| Best Practices for Administrators for PGCC | |
|---|---|
| **The Administrators for PGCC should:** | • Schedule Advisory Board meetings<br>• Meet with External Partners<br>• Provide supervision of Advisory Board Work Groups<br>• Provide monthly minutes for Advisory Board<br>• Provide an agenda for each Board meeting based on input from ITEC Coordinator at CMIT and at PGCC |

Table 3: Best Practices for Administrators for PGCC

In particular, several of the issues relating to concerns of all the stakeholders could be better addressed with a comprehensive professional development plan for the PGCC faculty and CMIT teachers. PGCC has over 25 years of experience conducting summer institutes for the PGCPS system. These institutes are typically interactive, engaging, and provide valuable pedagogy resources and teambuilding for faculty participants. College faculty members are rarely involved in teaching/learning interactions with high school students, and sometimes are unsure of how to engage them effectively to promote student success. Based on feedback from college faculty members who have taught in the College's existing dual enrollment program, professional development is crucial for success. Experienced PGCC dual enrollment faculty and faculty in PGCC's Teacher Education Department are now slated to develop and facilitate a training program to aid new ITEC faculty and teachers. The training program will include two days of instruction during the summer institute, class observations during their first year teaching in the program followed by immediate feedback from the professional development team, and two follow-up workshops during the academic year. An online learning community of dual enrollment faculty will be created using Blackboard so faculty can share ideas, resources, and challenges.

# 6 Conclusion

The dual enrollment model used for the AHS programs involves a partnership between the college faculty member and the high school teacher. For the past three years, PGCC and PGCPS have held a Dual Enrollment Workshop designed to introduce the faculty to each other, discuss their roles and responsibilities, and build the foundation for a solid team. A similar workshop structure will be scaled for ITEC and facilitated by experienced dual enrollment faculty, teachers, and PGCPS staff. The training will consist of a one-day workshop during the summer institute and a follow-up workshop prior to the start of each spring semester beginning in late spring of 2015.

With more of the recommended best practices in place, the evaluators are scheduled to duplicate the parameters of the present study by reassessing the ITEC program based on the implementation of the relevant practices and lessons learned for each stakeholder.

# 7 References

[1] Alvarez, C., Alex, D. E., & Harris, B. (2010). STEM specialty programs: A pathway for Under-represented Students into STEM Fields. *NCSSSMST Journal, 16*(1), 27-29.

[2] Born, T. (2006). Middle and early college high schools - providing multilevel support and accelerated learning. *New Directions for Community College,* 135, 49 - 58.

[3] Hurtado, S., Newman, C., Tran, M., Chang, Mitchell J., (2012). Improving the Rate of Success for Underrepresented Racial Minorities in STEM fields: Insights from a national project. *New Directions for Institutional Research, 148,* 5-15.

[4] Fleischman, S., & Heppen, J. (2009). Improving Low-performing High Schools: Searching for Evidence of Promise. *America's High School*, 105 – 133.

[5] Jackson, L., Lamar, C., Brown, Q., & Latson, V. (Fall, 2014). *Introducing the Big Ideas of Computer Science Through a K-12 TeacherProfessional Development Workshop.* Paper presented at ASEE MId-Atlantic Section Fall 2014 Conference, Swarthmore, PA.

[6] Jackson, L., Latson, V., & Gross, M. (2011) R*edesigning Core Programming Courses Through a Direct Instruction Approach.* Paper presented at International Conference Frontiers in Education: Computer Science & Computer Engineering, Las Vegas, NV.

[7] National Science Foundation, Division of Science Resources Statistics. (2011). Women, Minorities, and Persons With Disabilities In Science and Engineering. *National Science Foundation,* 11-309.

# Time-lapse Photography for K-12 Education

Mitchell L. Neilsen[1], James Shaffer[2], Nicholas Johnson[1]
Dept. of Computing and Info. Sciences[1], Dept. of Animal Sciences and Industry[2]
Kansas State University
Manhattan, KS, USA

**Abstract**

*This paper describes an innovative, new approach to combine visible, infrared, and thermal time-lapse photography from several different cameras to conduct empirical analysis in the K-12 classroom. This project is part of an innovative, new GK-12 STEM Fellowship Program which incorporates contemporary, embedded, real-time sensors and system design into the existing K-12 curriculum. Unlike other GK-12 programs, more focus is placed on Technology (T) and Engineering (E), and less focus is placed on Science (S) and Mathematics (M). The underlying goal of the program is to link cyber physical systems research with science and technology curriculum and create a community of learning, teaching, and mutual support between the higher and pre-college education participants.*

**Keywords:** Education, embedded systems, inquiry, outreach, real-time sensor networks.

## 1 Introduction

There is a well-recognized national need to inspire much more interest in Science, Technology, Engineering, and Mathematics (STEM) disciplines among K-12 students. Computational thinking is a type of analytical thinking that shares methodology used in other STEM disciplines [1]. Likewise, STEM disciplines have seen an increased emphasis on the use of computational thinking. Thus, it is important for us to increase the number of students from the K-12 level interested in pursuing STEM careers with an emphasis on computational thinking. When young students get excited about science and engineering as a result of experiences in school or informal education settings, they are more likely to pursue classes that properly prepare them for success in undergraduate and graduate programs in STEM fields. A primary goal of our GK-12 STEM Fellowship Program is to incorporate computational thinking into STEM disciplines in the K-12 classroom [2]. As noted by Shreck and Latifi, it is important to provide an infrastructure in which teachers have adequate support and where they are able to change with the new technology [3].

The U.S. National Science Foundation’s program to support Graduate Teaching Fellows in K-12 Education (GK-12) strives to improve graduate students’ communication, teaching, collaboration, and team building skills through professional training, interactions with faculty, and work in the classroom with K-12 teachers and students. By working with K-12 teachers to integrate their knowledge and research to enhance the classroom, graduate fellows and faculty members also have the opportunity to build partnerships with schools and teachers, and to enrich learning opportunities and increase motivation for K-12 students.

**Figure 1.** GK-12 STEM fellow helps students with sediment sensor

The graduate fellows also serve as role models for the students that they work with, and they talk with the students about the diverse and exciting careers that can be pursued by those who are interested in STEM disciplines. Although the main focus of the GK-12 program is on the development of graduate students, this paper will focus more on the innovative aspects of the program and some of the modules developed for use in the K-12 classroom. A detailed description of the time-lapse photography module will be given.

## 2 INSIGHT GK-12 STEM Fellowship Program

Infusing System Design and Sensor Technology in Education (INSIGHT) is the title given to our innovative GK-12 STEM Fellowship Program at Kansas State University. Our program focuses on integrating real-time embedded systems and sensor technology with computational thinking through a standards-based science, technology, engineering, and mathematics curricula.

The underlying goals of the program are to: enhance the usefulness, practicality and relevance of sensor, computing, and information technology education by linking embedded systems research for fellows to science and technology curriculum and to practice for classroom teachers; support technology in rural Kansas through two of the most important aspects impacting rural life in Kansas: agriculture and health; improve the teaching and learning of technology and engineering design in middle school and high school classrooms; and create a community of learning, teaching and mutual support between the both the higher education and pre-college education participants from rural backgrounds.

Project activities team GK-12 fellows with K-12 STEM teachers through summer and academic year training and orientation, and place the fellows in the classrooms of rural Kansas schools. In the summer, project staff provides fellows with training in hands-on sensor-driven systems, STEM concepts and development, Kansas Curriculum Standards, and classroom instruction methodology. Participating teachers also receive two weeks of training and orientation which focuses on sensor technology, computing and information science topics, selected science and technology content areas and the use of appropriate pedagogical and assessment strategies.

During the academic year, fellows support two participating teachers in the classroom an average of two times a week in their area with content-specific embedded sensor technology and computational thinking. Program staff provides semester-long professional development opportunities via guided research and investigations of practical applications of technology integration on agricultural farm fields and within the classroom. Weekly meetings between project staff and fellows provide supervision and feedback.

Sensor systems are poised to revolutionize the way that the physical world is monitored and field experiments are performed with remote, automated real-time data collection and feedback replacing traditional manual methods. The development of cyber-physical infrastructure represents the next step in enabling applications wherein physical entities (humans with body parameters such as heart and respiratory rates, crops with different fertility and growth rates, etc.) and cyber-subsystems collaborate and interact to achieve a common goal.

For example, in health-care systems, the cyber-infrastructure can augment the capabilities of the hospital staff in patient monitoring, issuing alerts, and coordinating usage of resources. Likewise, in rural Kansas, remote monitoring can enable elderly residents to stay in their own homes safely for an improved quality of life and an on-site pharmacist is replaced with a robot that can dispense prescriptions to elderly patients and allow patients to consult with a pharmacist remotely.

As another example, although farm equipment operators can operate with a local visual view of the field, cyber and remote sensing infrastructure in the field can assist them by providing correlated GIS, climatic and vegetation data to support variable rate application of chemicals with precision using GPS. This results in both economic and environmental benefits [4].

Typically, these systems are difficult to develop because their development requires knowledge about many parts of a complex system involving a number of heterogeneous subsystems and components. Their design is often a multidisciplinary exercise involving a variety of domain experts with different views of the system, and there are few formal techniques that can be used to address the integration of individual components. Designers often work on subsystems without fully understanding its impact on other components and the rest of the system. Design of such cyber physical systems has been the focus of researchers from several departments at K-State.

This program represents a unique synergistic opportunity for us to collaborate with our K-12 colleagues in a similar manner, and create and strengthen mutually beneficial partnerships with the many rural school systems in Kansas. These partnerships enhance the education of K-State's technologically-oriented graduate and undergraduate students while simultaneously advancing computing, science, and technology education in rural Kansas middle school and high school classrooms [5].

Sensor technology is the enabling element that pervades the entire science, engineering and technology curriculum, rather than as an entirely new and separate subject or curriculum area, whose introduction would be more problematic. Deductive reasoning, analysis and synthesis, algorithmic problem solving and design, and inquiry techniques are at the heart of each of these disciplines. Regardless of the scientific area, students must learn to formulate questions and hypotheses, plan experiments, conduct systematic observations, interpret and analyze data, draw conclusions and communicate results, using powerful classroom tools. Indeed, these skills are tested in a statewide assessment of students' achievement in science, engineering and technology. Aligning instruction with science, engineering and technology standards requires significant changes to classroom practice, from content, activities, and assessment to classroom management, interaction with students and learning tools. This program has helped to establish hands-on engineering and technology development as a foundational skill for vocational agriculture, science, mathematics, and other areas. Instead of focusing only on Mathematics and Science, the novelty of this project is on its primary focus on Technology and Engineering.

## 3 Sample Curricular Modules

In this section, we give a brief overview of a few modules that have been developed and/or delivered by fellows in the K-12 classroom. Details can be found on-line at, http://gk12.cis.ksu.edu, through our GK-12 program web-site. In addition, a detailed description of the time-lapse photography module is given below.

### *Water Filter*

In this activity students were asked to create their own sediment water filter using a water bottle and some basic materials. Some of the materials include: flour, sugar, sand, gravel, plastic beads, cotton balls, etc. The students were only allowed to use three materials and it was up to them to create the best filter in the class based on the types of materials selected, and order the materials were placed into the filter. The dirty water, shown in the upper-left corner in Figure 2, is stirred occasionally to keep the sediment suspended, and the sediment shown in the upper-right is used to test filtered samples.

**Figure 2.** Sediment water sensor and filters

### *Wireless Sensor Network to Measure Sediment*

At the high school level, students take samples using a hand-held sensor (Figure 1) and take readings using sediment sensors deployed in the field and connected by a three-tiered wireless sensor network [6-8]. A solar panel is used to power the middle tier of the wireless sensor network as shown in Figure 3.

**Figure 3.** Wireless sensor network

To measure sediment discharge, several turbidity sensors developed here at Kansas State University are organized into a wireless sensor network, and they continuously monitor sediment discharge. The system is organized to automatically adjust sensor reading rates based on the data to limit the power requirements of the wireless sensors. The data is then transmitted to a wireless base station, and on to a centralized database from which the data can be analyzed [6-8]. Sediment concentration is defined as the weight of suspended soil particles per unit volume of water. Turbidity is usually referred to as the optical properties of suspended or dissolved materials in water on transmitting, reflecting, absorbing, and scattering light. Thus, traditional turbidity sensors are not sediment-concentration sensors. A sediment sensor developed in this study uses LEDs that emit lights at three visible and infrared *feature wavelengths*, which were selected through a spectroscopic analysis, with light detectors arranged at different angles from the light sources. Statistical models established based on test data allowed the sensor to be basically insensitive to non-soil, suspended and dissolved objects, such as algae, organic matter, and various microorganisms, and less sensitive to soil texture. A prototype sensor was tested at combinations of four water types and five soil textures in the laboratory. Statistical and neural-network models successfully predicted sediment concentration across samples of all the combinations with $R^2$ values of no lower than 0.95. An outdoor experiment proved that the influence of ambient light on sediment measurement can be largely eliminated by modulating the lights. More than ten prototype sensors of different designs have been fabricated and calibrated.


**Figure 4.** Sediment concentration measurement

Several sensors were deployed at low-water crossings at Fort Riley in Kansas and Fort Benning in Georgia for long-term, sediment-runoff monitoring [7]. The sensor case has been modified to improve its waterproof capability. Difficulties encountered during the long-term tests included signal drifting and occultation of the optical lenses by algae and soil particles. Modifications in sensors and software have been implemented to solve these problems [6].

### *Water sediment concentration*

At the elementary level, sediment concentrations in water are measured manually using filter paper and drying equipment after collecting samples at a local lake. They also compared the results they obtained with the measurement taken using an electronic sediment sensor as shown in Figure 4.

### *Olympic bar acceleration during a bench press*

High school students in weight lifting classes at Wamego High School use Velcro to strap on a Wii remote to an Olympic bar to measure each student's acceleration in all directions while doing the bench press lift. A graph of the accelerometer values is projected onto the ceiling of the weight room so that students can watch their acceleration and movements during the lift. This information can be used to determine if the students are using correct lifting techniques.

**Figure 5.** Bench press acceleration

### *Time-lapse photography*

Middle school students analyze the effect of plant spacing on the growth rate of radish seedlings. Weekly measurements are made of average plant height and $CO_2$ levels in plant starter boxes. In addition to weekly measurements, we also record images of sprouting plants using inexpensive visible/IR cameras controlled by an inexpensive Raspberry Pi computer. Then, the images are automatically translated into a time-lapse video using a standard open-source media encoder called *mencoder* [14].

**Figure 6.** Time-lapse photography

At the beginning of the experiment, classes set up two groups of radishes. Groups consist of seven pots in an air-tight pan with a plastic lid. Holes are made 1.5" from the bottom of pans for use with a $CO_2$ probe (and subsequently taped over) and cameras are taped to the inside of lids. Treatment groups are either planted at a rate of one seed/pot or five seeds/pot. Students are shown connecting Raspberry Pi cameras to the inside of their pans in Figure 7.

**Figure 7.** Students connecting Pi cameras to plant boxes

All radishes are grown under a UV grow light for three weeks. Sprouting behavior is monitored by time-lapse photography during the first week of growth. Average plant height and $CO_2$ levels are measured at the end of each week for each box.

To make it easy for classroom teachers to set-up the cameras and specify experimental parameters, we have the Raspberry Pi's configured to look for a text file containing the specifications on a thumb drive and, if found, convert the text file to a script that is automatically executed to take pictures and store the images on the thumb drive. Then, the script is used to automatically generate a time-lapse video from the images stored. The steps to configure the Raspberry Pi systems are outlined below:

1. On Raspberry Pi, install the Raspbian GNU Linux 7 distribution and enable cameras [15].
2. Then, modify file /etc/rc.local, which is executed when the system is booted up, so that the system mounts the USB thumb drive under /mnt/usb and executes the commands in a script file called raspistill.txt or raspistill.py to capture the (.jpg) images and generate a video file (timelapse.avi) from the images generated, and then simply shut the Raspberry Pi system down. The original rc.local shell script file contains:

```
/etc/rc.local:
#!/bin/sh -e
#
# rc.local
#
# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
  printf "My IP address is %s\n" "$_IP"
fi
# ADDED CODE HERE
exit 0
```

We modify the file just above exit 0, as shown below, to add some code to tell the system to execute the commands listed in the script file on the thumb drive, for brevity we only show how to execute a shell script, but a comparison can be added to determine if a python script is available on the thumb drive as well:

```
# Added for time-lapse photography
mount -o uid=pi,gid=pi /dev/sda1 /mnt/usb
#
# Copy raspistill.txt to raspistill.cmd,
# convert to Unix format, and execute
cp /mnt/usb/raspistill.txt /mnt/usb/raspistill.cmd
dos2unix /mnt/usb/raspistill.cmd
sh /mnt/usb/raspistill.cmd
```

Some of the applications used are not installed by default, they can be installed them from the Internet using apt-get:

```
pi@raspberrypi $ sudo apt-get install dos2unix
pi@raspberrypi $ sudo apt-get install mencoder
```

Then, create a directory to be used to mount the USB drive, and set "pi" to be the owner of the directory:

```
pi@raspberrypi $ sudo mkdir /mnt/usb
pi@raspberrypi $ sudo chown pi /mnt/usb
pi@raspberrypi $ sudo chgrp pi /mnt/usb
pi@raspberrypi $ ls -l /mnt
drwxr-xr-x 2 pi pi 4096 Jan 14 19:30 usb
```

Then, we use any Windows editor to create an ASCII text file called raspistill.txt in the top-level directory on our USB thumb drive:

```
sleep 10
mkdir -p /mnt/usb/pictures
cd /mnt/usb/pictures
rm a*.jpg
rm timelapse.avi
raspistill -tl 5000 -o a%04d.jpg -t 30000
ls *.jpg > stills.txt
mencoder -nosound -ovc lavc –lavcopts
    vcodec=mpeg4:aspect=16/9:vbitrate=8000000 -vf
    scale=1920:1080 -o timelapse.avi –mf
    type=jpeg:fps=4 mf://@stills.txt
sync
sudo shutdown now
```

This example results in a picture once every 5000 ms (5 seconds) for 30 seconds. It's generally good to test a short running sample before you let it run for hours and hours. After 30 seconds, it will still take about 30 more seconds to create the video. Then, you can just turn off the Raspberry Pi, remove the thumb drive, and enjoy your video. If your time-lapse video includes hundreds of frames, it will take a bit longer to create the video, or you can create the video from the still images on a more powerful platform.

The Raspberry Pi can also be used to control the FLIR Lepton™ Thermal Imaging Camera. Simple scripts can be used to control the image capture and processing. Other scripting languages can be used to develop the scripts. Once the K-12 students gain an understanding of the simple scripting languages, they easily update the scripts to conduct their own experiments, and even write their own scripts. Since Python 2.7 is installed by default, to use python, just replace raspistill.cmd with raspistill.py, and invoke the shell by replacing the /bin/sh command with the python command: $ **python /mnt/usb/raspistill.py**. We also updated raspberry_pi_capture.c from Pure Engineering [11] to allow command line arguments to specify the output file name to enable the capture of numbered images for time-lapse photography. The new version is compiled as **rpcapture**, and incorporated into the inner-most portion of the python script as shown below.

```
# Jump to folder with visible/infrared images
os.chdir("/mnt/usb/normal")
# Take a visible/infrared pic of size 560x420
subprocess.call("raspistill -w
  "+str(imageWidth)+" -h "+str(imageHeight)+"
   -q 75 -o image"+str(i)+".jpg -t 50",
   shell=True)
# Jump to folder with thermal images
os.chdir("/mnt/usb/thermal")
subprocess.call("./rpcapture "+str(i),
   shell=True)
# Convert .pgm file created to .jpg and resize
subprocess.call("convert -resize
   '"+str(imageWidth)+"x"+str(imageHeight)+"!'
   thermal"+str(i)+".pgm thermal"+str(i)+".jpg",
   shell=True) …

def CombinePictures(leftPic, rightPic, number):
  newPic=Image.new("RGB", (cWidth, cHeight))
  left=Image.open("/mnt/usb/normal/"+leftPic)
  right=Image.open("/mnt/usb/thermal/"+rightPic)
  newPic.paste(left, (0, 0))
  newPic.paste(right, (secondPicX, secondPicY))
  newPic.save('combined'+str(number)+".jpg")
```

ImageMagick and Pillow can be used to perform image processing – converting .pgm to .jpg, and combining visible and thermal images into a single image to be included in the time-lapse video as shown in Figure 8 [12, 13]. This makes it a simple task to simultaneously capture both visible or infrared images using the Raspberry Pi camera port and thermal images using the Lepton FLIR Thermal Imaging camera connected to the GPIO pins.

**Figure 8.** Simultaneous visible and thermal image capture

To make it simple for K-12 students and teachers to focus on their experiments, and simplify the data capture and image processing, we just configured a master Raspbian OS image to have the additional software already pre-installed, and to have the modification already made to the rc.local file to automatically invoke a script if found on the thumb drive. The updated master image only needs to be copied onto an SD card for a new Raspberry Pi to be set up and ready to use.

## 4 Conclusions

This paper describes several new curricular modules for K-12 education. One module combines either visible or infrared images, with optional thermal images, for time-lapse photography. This module uses an inexpensive Raspberry Pi. It was developed as part of an innovative GK-12 STEM Fellowship Program that incorporates contemporary, embedded, real-time sensors, computational thinking, and system design into the existing K-12 curriculum.

A novel aspect of the time-lapse photography module is that K-12 teachers and students only need to edit a simple ASCII text file on a thumb drive to specify exactly the parameters to be used to capture the images and process the data retrieved. This empowers students to conduct their own empirical analysis in the K-12 classroom.

## Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. 0948019. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

## References

[1] J.M. Wing, "Computational thinking and thinking about computing", in Philosophical Transactions of the Royal Society A, Vol. 366, pp. 3717-3725, July 31, 2008.

[2] M. L. Neilsen, N. Bean, G. Singh, J. Spears, V. Wallentine, and N. Zhang, "INSIGHT: Infusing system design and sensor technology in K-12 education", in Proceedings of the 2012 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'12), 2012.

[3] S. Shreck and S. Latifi, "K-12 computer education deficiencies in Nevada", in Proceedings of the 2011 International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'11), pp. 114-118, 2011.

[4] M. N. Rao, D. Waits, M. L. Neilsen, "A GIS-based modeling approach for implementation of sustainable farm management practices", Environmental Modeling and Software, Vol. 15, pp. 745-753, 2006.

[5] M.L. Neilsen, N. Bean, and J. Spears, "Infusing cyber-physical systems into a standards-based K-12 curriculum", in Proceedings of the First Workshop on Cyber-Physical Systems Education (CPS-Ed), Philadelphia, PA, Apr. 8-11, 2013.

[6] N. Wang, N. Zhang, and M. Wang, "Wireless sensors in agriculture and food industry--Recent development and future perspective", Computers and Electronics in Agriculture, Vol. 50, No. 1, pp. 1-14, 2006.

[7] Y. Zhang, N. Zhang, G. Grimm, C. Johnson, D. Oarrd and J. Steichen, "Long-term field test of an optical sediment-concentration sensor at low-water stream crossings (LWSC)", *ASABE Paper No.* 072137, St. Joseph, Michigan, 2007.

[8] J. Wei, N. Zhang, D. Lenhert, M.L. Neilsen, M. Mizuno, and J. Schmidt, "Using smart transducer technology to facilitate precision agriculture systems", In Proceedings of the ASAE Annual International Meeting (ASAE Paper 03-3145), Las Vegas, NV, July 27-30, 2003.

[9] John Maloney, Mitchel Resnick, Natalie Rusk, Brian Silverman, and Evelyn Eastmond, "The Scratch Programming Language and Environment", *Transactions on Computer Education.* 10, 4, Article 16 (November 2010), pp. 16:1-16:15.

[10] M.L. Neilsen, D.H. Lenhert, M. Mizuno, G. Singh, J. Staver, N. Zhang, K. Kramer, W.J. Rust, Q. Stoll, M.S. Uddin, "Encouraging interest in engineering through embedded system design", In American Society of Engineering Educators (ASEE) Computers in Education Journal, Vol. XV, No. 3, pp. 68-77, July 2005.

[11] Lepton Engineering, LLC, "FLIR Lepton Module: raspberry_pi_capture.c", retrieved on 10/20/2014: https://github.com/PureEngineering/LeptonModule/blob/master/raspberrypi_capture/, 2014.

[12] ImageMagick Studio LLC, "ImageMagick Command Line Tools", http://www.imagemagick.org, retrieved on 10/20/2014 (use ***convert*** in python script).

[13] Fredrik Lundh, Alex Clark, and Contributors, "Pillow: Python Imaging Library (Fork) 2.8.1 for PI", https://pypi.python.org/pypi/Pillow/2.8.1, retrieved on 10/20/2014.

[14] Mencoder and mplayer official website and news: https://www.mplayerhq.hu/design7/news.html, retrieved on 10/20/1014.

# SESSION

# FLIPPING THE CLASSROOM AND RELATED SUBJECTS

## Chair(s)

### TBA

# The Coaching Role in the Flipped Classroom Results of a Study: First Year College Statistics

M. F Estabrooks
Red Deer College
Red Deer, Alberta, Canada

**Abstract:** *The proliferation of educational content on the internet continues unabated increasing by the hour exponentially. The benefits of this repository of information content are unquestionably profound and extremely useful to say the least. Yet the role and the degree to which this facility can be useful in a traditional educational setting may be questionable.*

*The author suggests that a blending of on-line content coupled with a classroom management scheme involving face to face interaction, student engagement scenarios as well as intervention techniques would alleviate many of the difficulties associated strictly on line learning. Such strategies would provide greater personalization of the learning process with the instructor providing a coaching role with a focus on the individual learner.*

*In this paper we present the results of a study of the success rates in college level statistics for a classroom situation using the notions of the flipped classroom where the instructor in effect performing a coaching role much as in the sports arena.*

## 1. Background

The Internet provides the capability to access information in a manner that is unprecedented in human history. With its interactive capability and the delivery of educational content in a 24-7 format, it promises that education and learning can now be designed to fit into the learner's time and space constraints and thus helps to ensure greater success in achieving his or her educational goals.

Yet the role of internet delivery of education is under fire in some circles, because of the high failure rates in some such courses and programs such as MOOC. The author suggests that this is due to the lack of proper support for students who do not have sufficient background knowledge of the subject matter but also who do are lacking proper study skills.

The author is of the opinion that extensive use of digital technology with its point and click technology does not lend itself to the development of an in-depth understanding of subject matter. On the other hand, pen and paper exercise and note taking combined with a human factor in general does tend to provide a much more in-depth processing of concepts and develop an understanding of the material. The author believes that the latter often involves the visual aspect of seeing ideas and schemes on paper (this may be referred to as doodling's) coupled with hand and musculature movement all of which tends to provide greater retention of concepts learned to the extent that these can be used to integrate concepts at a higher cognitive level.

In contrast to this, the author feels that an educational setting involving primarily on-line learning is often associated with a withdrawal of face to face (F2F) engagement leading to less retention and less meaningful learning outcomes. The dimensions of the learning is often very limiting in this context, as it often does not involve the multiple sensory mechanisms often associated with pen and paper methods and note taking. The author suggests that this aspect of on line learning affects the degree to which the typical learner becomes less engaged in the subject matter resulting in less than desirable learning outcomes.

If we add various other constraining factors such as addiction to social media, cell phone use, demands of work outside of school time as well as the demands and expectations of today's social setting, then we have a complex interplay of variables that is very likely to have significant impact on student success.

With these issues in mind, the author suggests a blending of internet delivery of content coupled with classroom management schemes that focus on the role of the instructor as a coach in interacting, monitoring motivating the student

This paper will present the results of study involving college level statistics at Red Deer College and efforts by the author to address the very diverse needs of students in several sections of the course.

## 2. A Proposed Blended Strategy

In an effort to address the issue of student success in today's technological environment, the author has identified several strategies to meet the diverse learning needs of the student in a classroom setting and provide greater opportunity for students to learn in his or her own time-space. Several aspects of this approach include:

- Learning Materials: Traditional mathematics and statistics textbooks often have intimidating page layouts filled with formulas and very tight text layouts. Moreover the coverage is often so vast as to be very daunting to the beginner level student.
- Presented with these constraints, the author developed a 266 page "class workbook" intended to provide a focus of activity for the student on a day to day basis. The workbook features key aspects of the theory behind statistical methods as well as numerous word problems designed to engage the student in the class. The page layout of the book provides space for students to demonstrate through hand written techniques the methodology to solve various types of problems.
- Textbook: Negotiations with a publisher resulted in the adoption of a customized text book with internet access to theory and word problems at a substantially reduced cost from the standard textbook. This enabled the student to carry out various on-line activities including study and guided inquiry in problem solving with on-line help as well as providing a means for self-testing. With appropriate on line feedback to the student, this scheme allowed the student to undertake a more or less independent approach to learning and allowing self-assessment to gauge one's skill level before the in class tests which were conducted on a quarterly basis.
- Videos: Over a period of time, the author developed 43 on line videos of 10 to 12 minutes duration, corresponding to 35 class hours. Each video addressed a topic that was covered in each class and these same questions appeared in the workbook thus enabling the student to do the workbook problems at home using these on line videos. The videos were made available in a 24-7 time frame in order that students are able to access the video content at any time.
- Weekly Guides: A detailed weekly study guide specifying which videos to watch, workbook problems to be completed, and on-line assignments to be done with timelines involved.
- Role of the Instructor: Key to the success of the strategies outlined above is the role of the instructor in the overall setting of "teaching and learning". As much as possible, instructor behavior in this setting was modelled after that of a coach, by way of face to face interacting, motivational behaviors and tracking student progress. Where it was felt necessary, various interventions took place where it was felt necessary. In some cases, the student might be asked to sign a particular declaration stating that s/he chose not to follow the directions or advice of the instructor as regards to behaviors or action relating to learning. Overall, the intent of the instructor behavior was to extract more of a commitment by the student to the course goals.

These strategies provided a setting for the study which is the subject of this paper

## 3. Implementation of the Strategy

In the Fall of 2014, two sections of beginner level college statistics was identified as part of the study. At the very beginning of each class, an effort was made to establish a relationship with the class and the student by way of outlining the structure of the course, the responsibilities of the instructor and the responsibilities of the student. Key to the success of the strategies was the face to face interaction between the instructor and student as well as student to student interaction.

A key aspect of the instructor's role was the ability to intervene in a student's life-space in order to remove what may be called "obstacles" to success. This could take the form of suggesting dropping a course or reducing the number of hours spent on outside work which would lead to greater chances of success in the course. This strategy proved to be very effective, but considerable effort was required on the part of the instructor to implement this aspect of the strategy.

Testing and feedback was considered an essential aspect of the strategy. Often the instructor would use a one sheet physical handout which would focus on problem solving aspect of the course. Often students were expected to work in groups to complete the handout during class time or at other times to be handed in the next class. This proved to be an excellent engagement tool for the class as a whole. During this time, the instructor took advantage of the time involved to interact with the students.

Noteworthy in this context was the extent to which students began to help each other in problem solving situations as well as making suggestions regarding which videos to view, which learning resources to use for study and so on. Overall, there appeared to be a greater sense of satisfaction with what was happening in the classroom. Indeed there were several occasions when a student would voice aloud that s/he was enjoying the classroom experiences.

## 4. Summary of the Results of the Study

Figure 1 below illustrates a histogram of the final percentage grades for the fall 2014 period.

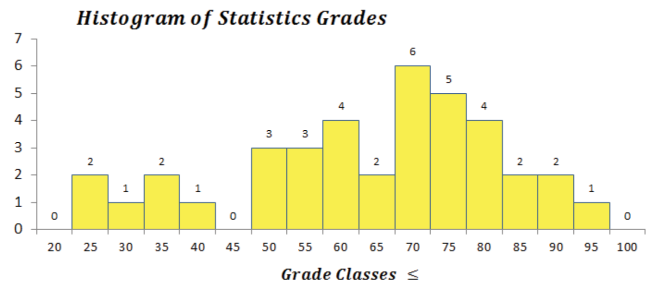



**Figure 1 Histogram of Final Grades Dec 2014**

Testing done at the beginning of the course revealed distributions which were multimodal in nature. In particular, the data revealed cluster points at 32, 45, 61 and 78. From a statistical perspective, this multimodal nature of the data suggests that several factors relating to student background were at play in the class.

Analysis of grades as the course progressed suggests that many students performing at the 40% to 60% level at the midterm of the course appeared to benefit from the instructor's teaching learning strategies to the extent that their grades moved upward to the 60-74% range. On the other hand, some of the students performing at the 35% to 55% achievement level appeared to have benefited very little from the strategies in place to increase student success. Indeed, some of these student's marks did become lower as the course progressed. This suggests that these students were immune to the efforts by the instructor to extract a commitment to the goals of the course.

Figure 1 above illustrates the bimodal nature of the final marks in the course. This observation suggests the presence of two distinct sub-populations. The author suggests that the instructor strategies were effective in those cases where the student was willing to make a commitment to the course goals and complete the work required for the course. However, where a student was not willing to make such a commitment of effort required to pass the course, it was inevitable this be reflected in the downward trend in his/her marks.

We suggest that the grade gap between those who pass and those who fail the course in such a learning environment becomes more pronounced as the role of the instructor as a coach, becomes more tuned to interaction, engagement and the tracking of student progress. It is suggested that the instructor may have a significant role to play in communicating his/her own expectations as regards learning outcomes of the students.

## 5. Instructor Role and Responsibilities

The implementation of the flipped classroom scheme results in a significant change in the role and behavior of the instructor. Most notable in this regard is the notion that the responsibility for learning in the course is more or less transferred from the instructor to the student. This observation suggests that the instructor spend less time in the delivery of content since the content was available on the internet. This meant that there was much less anxiety and stress on the part of the instructor for the actual delivery of content by way of board work and PowerPoint presentations.

In the role of the instructor as "coach", s/he is in effect relinquishing control of the delivery of educational content resulting in much more of a focus on activities and processes that enhance individual student outcomes. In this role, the instructor becomes more a "guide on the side" in contrast to a role such as "sage on the stage". The latter format is suggestive of a lecture based approach while the former being that of a coach in the educational setting.

## 6. Constraining Factors in Student Success

In an attempt to identify potential variables relating to student success or lack thereof, questionnaires were designed and administered by way of phone and/or personal interview methods as follow-up to the completion of the course. On the basis of the feedback obtained, the following "constraining factors" are considered significant by impeding student success:

- Lack of an appropriate mathematics background & appropriate study skills:
- Number of courses enrolled as well as number of lecture hours and number of lab hours involved in a program;
- Outside work: Some students would spend 16-24 hrs. each week working in bars, restaurants and office positions. Obviously, this is considered a severe constraining factor as regards to student success;
- Social media, internet & cell phone use: Attention devoted to such media can often be significant factors in limiting student success;
- Personal issues, family relationships, and issues relating to money also represent constraining factors;
- Questionable Planning and unrealistic assessment of time commitments to a course.

Obviously there are many factors at play in the context of our study and such factors may be considered to be of varying significance depending on the educational setting under scrutiny.

## 7. Changing Environment

The learning design schemes outlined in this paper have been developed and put to use over a period of several years and the results have been similar to those outlined here. Unfortunately, the success in recent years has been less dramatic. The lessening of the degree of success of the methods outlined above is due in part to the changing circumstances of the student. The following observations and speculations may be considered:

- Students less willing to assume responsibility for his/her learning in the course.
- Greater feeling of entitlement & expectations: It appears that and increasing number of students are of the opinion that “I paid for this course and it is your responsibility to get me through it”.
- Greater connectedness to cell phones and social media to the extent that these devices with their rapid response and continuous engagement features tend to capture students attention subverting attempts to focus on one line of thought for a short period of time.
- Difficulty with comprehension and understanding word problems, and an inability to develop hierarchical contexts in order to understand the various aspects of a problem;
- Students less willing to engage in face to face interaction, participation in discussion groups, or ordinary conversation, preferring to text message or involvement in social media.

With variables such as these at play in the wider context of the classroom, it is felt that multiple strategies may be utilized at any given time.

## 8. Conclusions

In the highly complex and connected social, technological and economic environment of today’s student, it is important that educators consider the complex interplay of the variables or “factors” associated with these environments as they impinge on students in the educational setting. In doing so, it is important to consider the instructor role as a key factor in the design and implementation of learning environments which focus on the individual student. In particular, the coaching role of the instructor/teacher with face to face interaction, engagement activity involving various human factors coupled with an appropriate balance of technology can play a significant role in increasing student success in a classroom setting.

But it is anticipated that these strategies themselves must be modified to varying degrees over time as we continually try to meet the changing needs of our students in these challenging times.

# Flipping a Data Structures and Discrete Mathematics Class

**Anja Remshagen**
Computer Science Department, University of West Georgia, Carrollton, Georgia, US

***Abstract** - This paper investigates the impact of the flipped classroom in a data structures and discrete mathematics course. The retention rates and students' grades when the flipped classroom model was applied in the course are compared with the retention rates and grades in the traditional lecture model. A two-week study was performed to compare the students' performance in the two different pedagogical models. Results are mixed and don't show a clear performance improvement in favor of the flipped classroom. However, grades and retention rates were significantly higher in the flipped classroom, indicating that the flipped classroom might help raising the confidence level of students.*

**Keywords:** computer science education, flipped classroom, traditional lecture, data structures, discrete mathematics

## 1 Introduction

Data Structures and Discrete Mathematics I (CS3151) is a required course to complete the Bachelor of Science in Computer Science at the University of West Georgia. The University of West Georgia is a regional, public institution. Most undergraduate students are traditional students that have started college right after finishing high school in the West Georgia region. Many of these students are first-generation college students. The racial demographics of the students is similar to the racial demographics of Georgia. For example, in the academic year 2013/2014, 33.5% of the students were Black/African Americans and 56.3% were Caucasion/White.

In the course CS3151, Data Structures and Discrete Mathematics I, computer science majors learn about elementary data structures and algorithms and about fundamental concepts in discrete mathematics. They implement and apply the data structures and related algorithms in small-scale Java programs. The course is typically taken by computer science majors after they have completed the introductory computer science sequence CS1 and CS2, which introduces students to the object-oriented programming language Java. As one of the first courses of theoretical and abstract nature, many students do not feel prepared for the rigor of CS3151. This is reflected by a high withdrawal rate. For example, in Fall 2012, out of 44 students 11 students withdrew from the class. In Spring 2013, out of 36 students 6 students withdrew. Some of the withdrawing students change their major; others attempt the class a second time.

To improve the retention rate in CS3151, the flipped classroom had been applied in Spring 2014. The flipped classroom is a pedagogical model where students learn new content on their own time outside the classroom and apply the material in problems and hands-on-activities during class sessions [5], [6]. The flipped classroom has been explored in several computer science disciplines, including introductory computer science courses [1], [7], computer architecture [4] and software engineering [3], to name only a few. Most studies demonstrate more engagement and enthusiasm among the students. Some studies show that there was no performance improvement [1], [4], or they had not yet sufficient data to study the students' performance in a flipped computer science course [3], [7]. This paper compares the students' performance in the flipped classroom with the performance in the traditional lecture model during a two-week study. It also explores retention rates and changes in grades among the recent course offerings.

## 2 The Flipped Classroom in CS3151

Before the introduction of the flipped classroom in CS3151, the class followed the traditional lecture model. Two issues that seem to impede students' success came up repeatedly when the traditional lecture model had been applied: (1) Students asked for more practice problems. (2) Some of the weaker students needed significant help to get started on the programming exercise. In Spring 2014, the flipped classroom was implemented to help students with the two issues addressed above and, in turn, to improve retention rates. As the current generation of students is used to consume visual media, video recordings for the course material were produced in the form of narrated slides, instead of assigning readings from a textbook. The students were required to watch the videos before class. In class, students worked through exercises individually or in teams of two. After working through one or two exercises, typically a student presented his or her solution. Sometimes the instructor worked out the solution with the students on the whiteboard.

As in preceding semesters, every week homework was assigned in Spring 2014. Most of the homework exercises were very similar to the exercises worked out in class and contained a mix of exercises at different levels of Bloom's taxonomy. In addition to written exercises, in many classroom meetings the students worked on a programming exercise that required the implementation or the application of a data structure and its associated algorithms. During the class meetings, students could program in teams of two. The programming exercises typically could not be finished in class due to time-constraints. Thus the completion of the programming exercises was assigned as part of the homework. Students had to complete all homework individually. At the end of the time reserved in class for the hands-on exercise, ideas of how to complete the programming task were discussed. This gave the students with weak programming skills a good start on programming exercises.

## 3 Study and Data Collection

In order to determine whether students had a better understanding of the material, the concepts, and of the algorithms by using the flipped classroom, the class was split into two groups for one week. Group L was exposed to the traditional lecture model during that week, and Group F to the flipped classroom. Both groups were taught exactly the same material. Group L met on a Monday for a traditional lecture that introduced the abstract data type priority queue, the heap data structure, its associated operations, and heap sort. The same material that was covered in the lecture was recorded through narrated slides. The recordings were assigned to Group F to watch before the class meeting on Wednesday. The recordings were not available to Group L. Group F worked through exercises in their class meeting. A pre- and posttest assessed the progress of each group. Table 1 displays the schedule for this study.

**Table 1. Schedule of interventions.**

| | |
|---|---|
| Wednesday (Mar 24) | Pretest taken by Group L and F<br>Assignment of videos only accessible to Group F |
| Monday (Mar 31) | Lecture for Group L<br>Assignment of homework to Group L |
| Wednesday (Apr 2) | Flipped classroom meeting for Group F<br>Assignment of homework to Group F |
| Monday (Apr 7) | Posttest taken by Group L and F<br>Homework due for Group F and Group L |

Different problems were tackled by the students in examples during the lecture, in exercises for the flipped classroom, in a homework assignment, and in the pre- and posttest. We can split all these problems into four categories outlined in Table 2. The first category (KNOW) required students to understand concepts and recall related facts. These problems are located at the first and second level of Bloom's Taxonomy. The second category (PRP) required students to interpret the representation of heaps and heap properties. These problems are at the second and third level of Bloom's Taxonomy. The third category (ALG) demanded students to perform and trace the algorithms that are executed by the operations on priority queues and heaps. These problems are at the third level of Bloom's Taxonomy. The last category

**Table 2. Problem categories.**

| Category | KNOW | PRP | ALG | OTHER |
|---|---|---|---|---|
| Description | General knowledge about heaps and priority queues | Properties of heaps and their representation | Operations on heaps and priority queues | Includes problems at the Bloom's synthesis level |
| Examples in lecture for Group L | | x | x | |
| Exercises in class for Group F | | x | x | x |
| Homework for Group L and F | | | x | x |
| Pre/Posttest for Group L and F | x | x | x | |

(OTHER) included problems at higher Bloom levels. These problems were not suited for a quiz-like test that can be completed in a few minutes. Thus only the first three categories were assessed in the pre-and posttest. Table 2 displays the occasions at which the students had been exposed to problems of the corresponding categories.

Students in Group L were occasionally asked problem-solving questions during the lecture, but participation was voluntary. The recorded slides for Group F did not contain all the examples that were worked out on the whiteboard as part of the lecture. Instead, Group F worked through related exercises on Wednesday. In addition to problems of category ALG and PRP, Group F worked through a problem of category OTHER.

To assess the difference in performance, a pretest and posttest were administered in the class meeting in the week before, respectively after, the groups met separately. The pre- and posttest contained eight questions as shown in Table 3. The questions are split into the three categories KNOW, PRP, and ALG.

The goal of the assessment is to determine whether students learn better through the use of the flipped classroom. In the lecture model learning takes place mostly outside of the classroom. In CS3151, many students delay studying the class material until they have to complete a homework assignment. Hence to fairly compare the learning in both models, all students had to complete the same homework assignment before the posttest was given. The assignment consisted of two problems of the category ALG and of two problems of category OTHER. Hence not only the students in Group F, but also the student in Group L, that followed the lecture model, had worked on exercises themselves as well. Programming exercises were not assigned in that week.

**Table 3. Problem categories.**

| | | |
|---|---|---|
| KNOW | Q1 | Determine whether a heap is a suitable data structure to implement a waiting line in discrete event-driven simulation. |
| | Q2 | Identify the concept that underlies heap sort. |
| | Q3 | Identify the time complexity of heap sort. |
| PRP | Q4 | Determine the arrays representation of a heap. |
| | Q5 | Determine whether a given tree is a heap. |
| ALG | Q6 | Trace a given sequence of operations on a priority queue. |
| | Q7 | Trace the insert operation on a given heap. |
| | Q8 | Trace the remove operation on a given heap. |

## 4 Results

Table 4 shows the results of the pre- and posttest. The maximum points for each question were 1.25. The mean and the standard deviation is displayed for each test question for Group L and F. The column labeled DIFF displays the difference among the means of the pre- and posttest.

**Table 4. Results of the pre- and posttest.**

| | | Group L | | | | | | Group F | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Pretest | | Posttest | | Diff | ES | Pretest | | Posttest | | Diff | ES |
| | | MEAN | SD | MEAN | SD | | | MEAN | SD | MEAN | SD | | |
| KNOW | Q1 | 0.63 | 0.66 | 0.25 | 0.53 | -0.38 | -0.44 | 0.42 | 0.65 | 0.42 | 0.65 | 0.00 | 0.00 |
| | Q2 | 0.88 | 0.60 | 1.00 | 0.53 | 0.12 | 0.16 | 0.42 | 0.65 | 0.83 | 0.65 | 0.41 | 0.46 |
| | Q3 | 0.50 | 0.65 | 0.63 | 0.66 | 0.13 | 0.14 | 0.00 | 0.00 | 0.42 | 0.65 | 0.42 | 0.65 |
| PRP | Q4 | 0.00 | 0.00 | 1.13 | 0.40 | 1.13 | 2.85 | 0.21 | 0.51 | 0.63 | 0.68 | 0.42 | 0.49 |
| | Q5 | 0.41 | 0.49 | 1.13 | 0.40 | 0.72 | 1.14 | 0.42 | 0.65 | 1.20 | 0.127 | 0.78 | 1.19 |
| ALG | Q6 | 0.50 | 0.65 | 1.13 | 0.40 | 0.63 | 0.83 | 0.83 | 0.65 | 1.04 | 0.51 | 0.21 | 0.25 |
| | Q7 | 0.25 | 0.53 | 1.13 | 0.40 | 0.88 | 1.33 | 0.21 | 0.51 | 1.04 | 0.51 | 0.83 | 1.15 |
| | Q8 | 0.25 | 0.53 | 0.63 | 0.66 | 0.38 | 0.44 | 0.42 | 0.65 | 1.04 | 0.51 | 0.62 | 0.76 |

According to Table 4, the lecture group showed gains in some areas while Group L showed gains in others. We discuss the results for each category. Both groups did not perform very well on the questions in the first category (KNOW). This is not very surprising, as the questions in this category had not been practiced in the homework or in one of the class meetings in any form. Group F shows a stronger improvement than Group L. It seems that the students were better able to retain the knowledge through the posted videos than through the lecture. Possibly, the students in Group F profited from the advantage that they could review the videos repeatedly if they could not follow the presented material the first time.

Recall that the lecture group had seen examples and the flipped classroom group had worked through exercises related to the questions in the second category (PRP). But there had been no homework exercises on the corresponding topic. Group L improved significantly in the first question of category PRP, while Group F improved only slightly. Both Groups showed a similar improvement in the second question, where the improvement was slightly stronger for Group F.

In the last category (ALG), Group L improved more than Group F in the first two questions, and Group F improved more in the last question. Recall that both groups had seen examples and exercises, respectively, of the corresponding topic. Both groups had seen questions in the homework assignment that were very similar to the questions in the pre-and posttest. These questions practice operations on priority queues and heaps. The complexity of the operations in Q6, Q7, and Q8 increases with Q6 covering the least complex operations and Q8 covering the most complex one. It seems that the hands-on exercises in the flipped classroom helped the students learn in particular the more complex the problems better. Another reason for the better performance on complex tasks might be that students in the flipped classroom had gone through the correct solution to similar exercises in the class. For simpler exercises most students are able to determine the correct solution on their own. But for complex ones, students may need to see more solutions of related problems.

We also compare the final grades of students in Spring 2014 with the grades in preceding semesters when the traditional lecture model had been applied. Before Spring 2014, CS3151 had been offered in Spring 2013, Fall 2012 and Fall 2011 and had been taught by the same instructor as in Spring 2014. There has been significant improvement concerning the withdrawal and passing rates. In Spring 2014, 81.5% of the students passed the class with a C or better. The passing rates in the preceding semesters were 66.7% (Spring 2013), 63.6% (Fall 2012), and 53.7% (Fall 2011), respectively. However, it is hard to compare final grades as the grade calculation has differed in different semesters. For example, the grades from Fall 2011 to Spring 2013 included a quiz grade that was determined by the use of iClickers. Also the grading rubric for homework had been changed in Spring 2014, typically resulting in higher grades. Thus, it is hard to determine if the students' performance has indeed improved, and if so, in how far the improvement is based on the flipped classroom model.

Table 5 lists for each semester and for each final letter grade the number of students achieving that letter grade and the corresponding percentage of students with that letter grade in the class. For example, in Spring 2014 four students received an A in the class, which is 14.8% of the students. This is the highest percentage of students that have earned the letter grade A compared with the other class offerings. No student withdrew from the class in Spring 2014 while about 17% to 25% withdrew from the class in preceding semesters.

**Table 5. Grade distribution per semester.**

| | Fall 2011 | Fall 2012 | Spring 2013 | Spring 2014 |
|---|---|---|---|---|
| Class size | 54 | 44 | 36 | 27 |
| A | 3 (5.6%) | 1 (2.3%) | 2 (5.6%) | 4 (14.8%) |
| B | 14 (25.9%) | 15 (34.1%) | 5 (13.9%) | 6 (22.2%) |
| C | 12 (22.2%) | 12 (27.3%) | 17 (47.2%) | 12 (44.4%) |
| D | 10 (18.5%) | 4 (9.1%) | 2 (5.6%) | 2 (7.4%) |
| F | 4 (7.4%) | 1 (2.3%) | 4 (11.1%) | 3 (11.1%) |
| W | 11 (20.3%) | 11 (25.0%) | 6 (16.7%) | 0 (0%) |

Considering that no student withdrew from CS3151 in Spring 2014, the flipped classroom seems to strengthen the students' confidence level, which has also been confirmed by [2], for example. One might argue that the higher grades might be the cause for the low withdrawal rate. Probably the grades of a student play an important role in the decision to withdraw from a course. But this is likely not the only cause since in previous semesters many student withdrew from the course even though they still had passing grades.

As pointed out before, the letter grade does not offer a true comparison as far as mastering the course content is concerned. A better indicator whether students perform better might be the test grades. In all semesters very similar tests were given. Table 6 lists the total average test grade. In each semester, except for Fall 2011, three tests were given. In Fall 2011, four tests were given. The maximum test grade for each test is 100. The test grades do not demonstrate a clear performance improvement. Note the total average includes only the test grades of students that had not withdrawn from the course. Thus the calculated average may favor semesters at which more students withdrew.

**Table 6. Total average tests grades per semester.**

| Semester | Average total test grade |
|---|---|
| Spring 2014 | 72.07 |
| Spring 2013 | 67.81 |
| Fall 2012 | 73.95 |
| Fall 2011 | 71.18 |

## 5 Conclusions

Based on the two-week study and on test grades, it does not seem that students are performing better overall on exercises in the flipped classroom compared to the traditional lecture model. The performance was compared after the corresponding material had been covered and practiced in both classroom models. There is some indication that the students perform better in the traditional classroom on less complex problems for which they have seen examples in class. Students may perform better on complex problems in the flipped classroom if those problems have been practiced in class. The final grades have been significantly better in the flipped classroom. However, a comparison among different semesters is difficult due to the different student population and the difference in grade calculation. No student withdrew from the flipped classroom while about 17% to 25% of the students had withdrawn from previous course offerings that applied the lecture model. This may be contributed to the higher grades students received before the withdrawal deadline. But the flipped classroom may also raise the students' confidence level as not all withdrawals are made by students with failing grades.

## 6 References

[1] Campbell, J., Horton, D., Craig, M., Gries, P., Evaluating an Inverted CS1, *Proceedings of the 45th ACM Technical Symposium on Computer Science Education*, 307-312, 2014.

[2] Enfield, J., Looking at the Impact of the Flipped Classroom Model of Instruction on Undergraduate Multimedia Students at CSUN, *TechTrends,* 57 (3), 14-27, 2013.

[3] Gannod, G. C., Burge, J. E., Helmick M. T., Using the inverted classroom to teach software engineering, *Proceedings of the 30th International Conference on Software Engineering*, 770-786, 2008.

[4] Gehringer, E. F. & Peddycord III, B. W., The Inverted-Lecture Model: A Case Study in Computer Architecture, *Proceedings of the 44th ACM Technical Symposium on Computer Science Education*, 489-494, 2013.

[5] Hughes, H., Introduction to Flipping the College Classroom, In *World Conference on Educational Multimedia, Hypermedia and Telecommunications*, 2012 (1), 2434-2438, 2012

[6] Lage, M. J., Platt, G. J., Treglia, M., Inverting the Classroom: A Gateway to Creating an Inclusive Learning Environment, *Journal of Economic Education,* 31 (1), 30-43, 2000.

[7] Lockwood, K., Esselstein, R., The Inverted Classroom and the CS Curriculum, *Proceedings of the 44th ACM Technical Symposium on Computer Science Education*, 489-494, 2013.

# Integrating Flipped Classroom Approach and Work in Pairs into Workshops in Programming Course

**Marina Lepp and Eno Tonisson**
Institute of Computer Science, University of Tartu, Estonia

**Abstract –** *Active learning approaches have been suggested by many academics for improving students learning to program. This paper describes an attempt to integrate flipped classroom, collaborative learning and peer review approaches into workshops in a compulsory first-year object-oriented programming course. The students in our course have to read the workshop material and solve some problems outside the class. In the workshop, the students work in pairs, and review and correct the homework of each other together. Students were surveyed to obtain their views on the work in pairs and flipped classroom approach in the workshops in the programming course. Results indicate the student's positive attitude to such organization of workshops.*

**Keywords:** flipped classroom, collaborative learning, work in pairs, peer review, programming learning, workshops.

## 1 Introduction

Learning to program can be very hard [10]. Different approaches are used to make learning easier. Lately, much attention is paid to using student-centered active learning approaches instead of traditional teacher-centered approaches. Different techniques are presented to support student learning using student-centered approaches. Flipped classroom and collaborative learning are some of them.

The flipped classroom approach, introduced in 2000 by Baker [3] and Lage et al. [11], means that events traditionally taking place inside the classroom are organized outside the classroom and vice versa. This approach is usually applied to lectures where students watch video lectures at home, and discuss and solve the exercises in class. However, Bishop and Verleger found in their survey of research [5] that "students tend to prefer in-person lectures to video lectures, but prefer interactive classroom activities over lectures". The present paper describes an attempt to integrate the flipped classroom approach into workshops, leaving lectures in the class, in a compulsory first-year object-oriented programming (the second programming) course at the University of Tartu, Estonia. Traditionally workshops mean that students read the materials (tutorials) and solve the problems weekly in the class and teachers correct the problems and give feedback after class. This is what we did for years. As the groups are quite large (usually we have around 200-250 students) the correction and giving feedback can take a lot of time. Some courses are using automated assessment systems (like VPL, the Virtual Programming lab for Moodle [16]). We wanted to engage students in this process.

Students' engagement in the learning process is one of the key moments in active learning [15]. There exist different types of active learning, like peer-assisted learning, collaborative learning, cooperative learning and problem-based learning. We use a mixture of collaborative learning and peer review. The students in our course have to read the workshop material and solve some problems alone outside the class (flipped classroom). In the class, the students work in pairs (collaborative learning), and review and correct the homework of each other together (peer review). In addition, they fill out the worksheet with questions about programs. After that, the students solve additional problems.

This paper describes the organization of the course and its history (how we came to this solution). There are two specific research questions of interest. What do the students think about the work in pairs in the programming workshop? What do the students think about flipped classroom in the workshop? Both of these questions are answered using a survey of students taken in 2015. In addition, we asked the instructors of workshops for their opinion on this organization of workshops.

The paper has the following structure: Section 2 gives an overview of related works, in particular in using flipped classroom and active learning approaches. The organization and history of the course and conduct of the survey for answering the research questions are thoroughly described in Section 3. The results of the survey are provided in Section 4. Section 5 concludes the paper.

## 2 Related works

Reviewing the literature, Andrews et al. [2] found that "there is a consensus among education researchers that much of the difficulty students have learning science can be attributed to the passive role students play during traditional lectures". They also found that active learning could improve learning. They said: "A classic example of active learning is a think–pair–share discussion, in which students think about a question posed by the instructor, pair up with other students to discuss the question, and share answers with the entire class." A very good overview of active learning is provided by Prince [15]. Prince defined common forms of active learning, which are most relevant for an engineering faculty, and examined the evidence for effectiveness of active learning. Prince defined active learning as "any instructional method that engages students in the learning process" and described, "active learning requires students to do meaningful learning activities and think about what they are doing". The definition of collaborative learning provided by Prince is the following: "Collaborative learning can refer to any instructional method in which students work together in small groups toward a common goal". Prince's study has found support for all forms of active learning examined and has presented the benefits of student engagement. Likewise, Michael [13] found that "there IS evidence that active learning, student-centered approaches to teaching physiology work, and they work better than more passive approaches".

Active learning approaches to teaching and learning programming are also under discussion and concern in different papers. Barg et al. [4] have designed problem-based learning (PBL) courses (one of them was Introduction to Programming) and have found that a larger proportion of trial students expressed a positive attitude to learning and a positive feeling about the course. O'Kelly et al. [14] found: "An intangible effect that we have observed is the support mechanism that has resulted from the PBL workshops. Through these workshops, students had the opportunity to get to know each other better, and to recognise how other students approach and represent a problem. We believe that this has been of significant benefit to the students." Zhang et al. [22] suggested that, "when teaching introductory programming courses, instructors may want to consider choosing the student-centered active learning over the traditional lecture format in order to improve students' learning performance".

Other very useful active learning approaches include peer assessment and peer review. Topping [20] provided a review of peer assessment and defined peer assessment as "an arrangement in which individuals consider the amount, level, value, worth, quality, or success of the products or outcomes of learning of peers of similar status". Many previous efforts to apply peer assessment and peer reviewing in computer science have involved web-based (online) peer assessment – reviewing tools. Sitthiworachart and Joy [17] described peer assessment as "a method of motivating students, involving students discussing, marking and providing feedback on other students' work". After developing and evaluating a web-based peer assessment tool for an undergraduate programming class, they found that "computer-mediated peer assessment is a valuable assessment approach which promotes active learning". Wang et al. [23] developed an online assessment system called EduPCR and after using it found that student learning in various aspects had significantly improved. Søndergaard and Mulder [18] examined student peer review rather than peer grading and listed the benefits also for the feedback provider with special consideration of computer science and software engineering. They concluded, "student peer assessment is a worthwhile exercise". Turner et al. [21] described the positive effects of peer code review in teaching object-oriented programming.

A quite recent active learning approach is flipped (also called inverted) classroom approach, which was introduced in 2000 by Baker [3] and Lage et al. [11]. In the broadest sense, flipped classroom represents a re-ordering of classroom and at-home activities. Bishop and Verleger [5] provided a survey of the research of flipped classroom. They proposed that flipped classroom actually represents an expansion of the curriculum, rather than a mere re-arrangement of activities, as group-based and open-ended problem solving is usually added in flipped classroom. They also reported that student perceptions of flipped classroom are somewhat mixed, but are generally positive overall. Gannod et al. [9] found that the response of the students from a software engineering curriculum to the course was overwhelmingly positive, especially in regard to the use of the inverted classroom model. Butt [7] found that 75% of total respondents in a final-year course in the undergraduate actuarial program viewed flipped classroom as being beneficial to their learning experience compared to a didactic lecture structure. Conversely, Strayer [19] compared the learning environments of an inverted introductory statistics class with a traditional introductory statistics class at the same university and found students in the inverted classroom were less satisfied with how the classroom structure oriented them to the learning tasks in the course, but they became more open to cooperative learning and innovative teaching methods.

Abeysekera and Dawson [1] provided a call for research as they found that the flipped classroom approach is under-evaluated, under-theorized and under-researched in general.

# 3 Data and methodology

## 3.1 Course background

This article describes an object-oriented programming course for first-year undergraduate students at our university. This course is the second programming course and is taught during the second (spring) semester. The course is compulsory for all computer science students (around 150 students every year), as well as for mathematical statistics students, who select the computer science module. However, this course is also popular among physics and mathematics students. As a result we have around 200-250 students.

The prerequisite for the course is a course titled "Programming", which has been based on Python in the last 6 years [12], where the basic concepts of programming are introduced to students. The object-oriented programming course is based on Java. The course consists of two forms of teaching: lectures and workshops. The students have one lecture (90 minutes) and one workshop (90 minutes) per week for 16 weeks. The first author of the paper has been involved in teaching the workshops since 2006. The second author is teaching lectures since 2012 and workshops since 2006.

Traditionally, all students of the course come together for lecture in one classroom. As we have quite a large number of students (in 2015 we have 198 students registered for our course) their background and abilities are quite different. This spring we offered three different formats of lectures for students to choose. Most students participate in traditional lectures, where the lecturer gives the lecture in front of the class. During the lectures we use clickers (student response systems) as an active-learning component to increase classroom engagement and motivation (it was found that clickers are suitable for engaging students in classroom discussion, especially in large classes [6]). The lectures are videotaped and these videos are used in the second format, which is called “nonstationarity” group. In this group, the students have to watch the video outside the class and answer some questions about the lecture. About 20 students chose the “nonstationarity” lecture format. The third format of lectures is called “intensive” group and is intended for students who are already quite good in programming. Their lecturer covers the compulsory material at a faster pace and after that they have time to study additional and more difficult material. Around 20 students preferred this format of lectures.

We did not use flipped classroom for the lectures because of the large number of students. We think it is very hard to interact with around 150 students in the class. Gannod et al. [9] found that “for larger classes, providing the desirable amount of instructor-student interaction would necessitate breaking these classes into smaller sections”. They also proposed that flipped classroom may not be suitable for a traditional large lecture course involving hundreds of students.

## 3.2 Organization of workshops

Workshops play a big role in the object-oriented programming course as students can earn 54% of final grade during workshops (programming tests, homework, projects). Homework with work in pairs during workshop gave 12% of the grade. Workshops are conducted in smaller groups with around 20 students in one group with the help of several instructors. In 2015, we have seven instructors guiding the ten workshop groups.

The organization of workshops has been very variable over time. For years, we used a traditional approach, where students came to the workshop and had to read the tutorial and write some programs. The students were very passive as they were reading most of the time. The reading did not lead to a lot of question. The tutorials were quite long, with explanations and examples, and the students did not have enough time to program. The students usually had to finish their programming exercises at home. We used different ways for checking the programming exercises that were finished at home. One way was for the students to upload the programs to a web environment (e.g. Moodle) and the instructor left comments about the programs for the student there. This approach does not suit us very well, because we cannot be sure that the student himself solved the problems. Then we tried to check the programs during the subsequent workshop. The students had to come to workshop, to show the programs and to explain them. This was a good approach; the instructor had an overview of every student’s skills and abilities. However, there was one problem – the instructor did not have time to explain or to answer questions. The whole time during a workshop, the instructor moved from one student to another and checked the programs.

In 2013, we decided to try the flipped classroom approach. Now the students have to read the tutorial and solve some problems before class. To solve the problem with checking the students programs and giving feedback we decided to use peer review and work in pairs. We prepared a worksheet for every workshop with questions about programs and students have to present their own solutions, to check a peer’s program and to answer the questions about the peer’s program. The students have to correct and improve the programs of each other together. During that time the instructor walks around in the classroom and looks at the students’ programs and answers as well. After filling in the worksheet, the students have time to solve additional problems prepared for workshops. They can solve together in pairs, but they can also program separately, only asking questions from the peer and the instructor. The instructor can also solve some problems with the whole class. This means that students get more practice of programming due to flipped classroom as Bishop and Verleger [5] proposed.

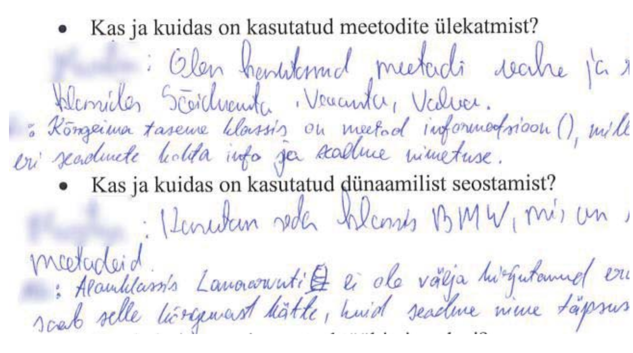



Figure 1. A part of the worksheet with students’ answers

Figure 1 shows a part of the worksheet with the students’ answers. The topic of this workshop was superclasses and subclasses. The first question is “Whether and how the methods overriding was used?” The student answers are: “The overriding is used for methods difference and toString in classes Car, Truck and Volvo” and “Superclass has method information(), this method is overridden in subclasses”. The second question is “Whether and how the dynamic binding was used?" The students wrote: "It is used in the class BMW, which is empty and is using methods from class Car" and "There is no method information in the class Computer. The class will use the method from superclass, but there will not be the specific information".

The crucial question in organizing work in pairs is how to form pairs. One possibility is to let the students form pairs. We do so during some workshops (e.g. when the students have to present their projects). However, usually the students select their friends as peers. As they do the project with a peer selected by them (usually one and the same friend) and they can also discuss the homework with the friend before class, it would not be very beneficial to see the solution of this friend. We think the students will benefit if they can see how different people think and solve the problems. That is why we form the pairs randomly. In order to make the process more fun for the students, we use different games to form pairs, like memory game, 2-pieces puzzles, playing cards, etc.

This is our proposed scheme of the workshop, but we have different instructors guiding the workshops. Of course, different instructors conduct the workshops in a slightly different manner. However, the main components (flipped classroom, peer review and work in pairs) are presented in workshops of all instructors.

### 3.3 Data collection

In 2015, we decided to organize a survey to investigate what the students think about our workshops. The survey had two parts with questions. The first part was about work in pairs. It had 3 multiple-choice questions about work in pairs in general and 15 questions for answering using a 5-point Likert scale (ranging from a score 1 ("strongly disagree") to 5 ("strongly agree")) about work in pairs in our workshops. The second part of the survey concerned the flipped classroom approach. It had 4 general multiple-choice questions and 25 questions with the same Likert scale about the use of flipped classroom in our workshops. In addition, the survey had two questions with free text answer about what the students like the most or dislike the most about the workshops. A field for general comments was provided as well.

The questionnaire was administered to students during one lecture in March 2015. The students who participated in the traditional lecture or in the "intensive" group were given the questionnaire on paper. A Google form with the same questions was provided to the "nonstationarity" group and to those who missed the class. The online form was made available for a one-week period. A total of 160 students answered the survey, representing a response rate of 81% of all 198 students.

In addition, we asked the instructors of the workshops for opinion on the organization of the workshops. They had to answer ten questions about their workshops.

## 4 Results

Data obtained from the survey (questions with Likert scale) were analyzed using Cronbach's alpha to determine the internal consistency of the responses [8]. Cronbach's alpha on 160 responses for the questionnaire was 0.771, which suggested that the survey tool had a good level of internal consistency and reliability.

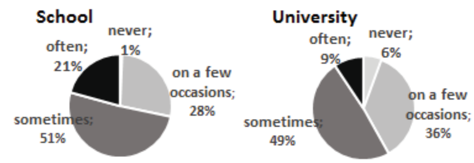



Figure 2. Student participation in work in pairs in school and in university

The students are familiar with work in pairs (Fig. 2) as quite many students indicated that they had worked in pairs often (21% in school and 9% in university) or sometimes (51% and 49%, respectively). The students seem to like rather than dislike the work in pairs (left part of Fig. 3). 49% of the students like the work in pairs, 28% selected the option "neither like nor dislike" (many of them marked in the comments that pleasantness depends on partner) and only 24% of students marked that they tend to dislike work in pairs.

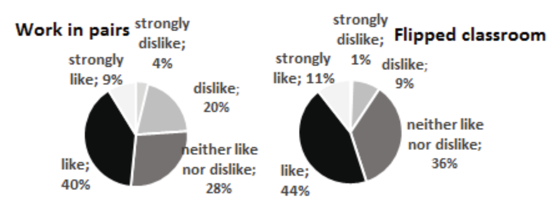



Figure 3. Pleasantness of work in pairs and flipped classroom to students

Fifteen statements from the survey about work in pairs in the workshops are shown on Figure 4. The statements were formulated using the word "usually", because pairs are formed randomly every workshop and particular peers can sometimes be unsuitable or it can happen that the work does not go well, even though usually this is not the case.

This study demonstrates that around half of learners seem to agree that work in pairs has a positive effect. The work in pairs usually goes well (statement 2) and paired students usually are suitable for collaborative work (statement 3). Half of the students think that the worksheet encourages discussion (statement 11) and two thirds of the students agreed that discussion with a peer is instructive (statement 4). Half of the students marked that work in pairs takes more time than working alone (statement 5), but the students agreed that they get more knowledge and skills thanks to working in pairs (statement 7), they get advice and explanations from the peer (statement 8) and the outcome of the work becomes better (statement 6). The students could not indicate whether they or their peers have more knowledge and skills (statement 9) and who plays the leading role in discussion (statement 10). The responses to the question about having more time for working alone were quite varied (statement 14). The students disagreed that both peers should have separate worksheet (statement 13). Overall, the students seem rather to like reviewing the homework in pairs (statement 1).

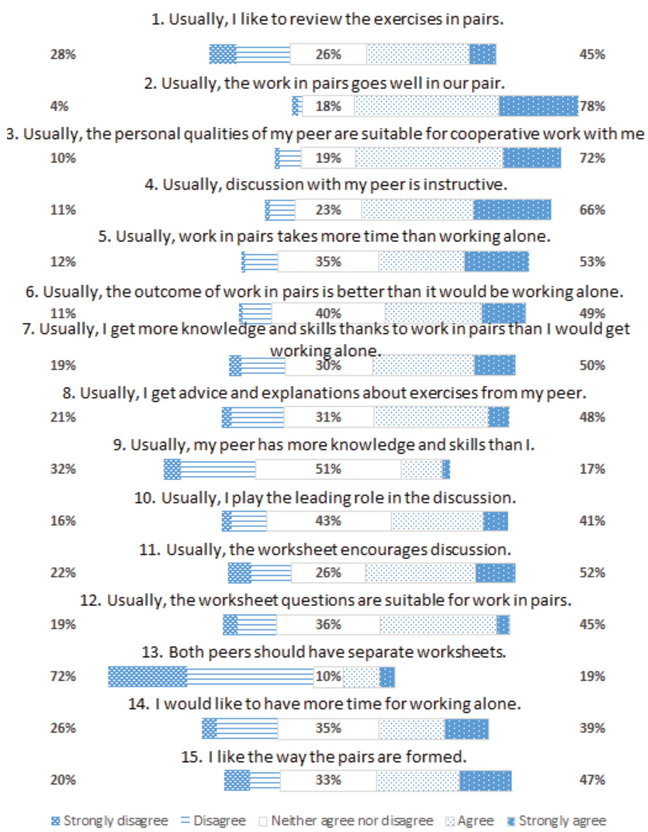



Figure 4. Likert responses to items connected to work in pairs

It seems that the flipped classroom approach is not very common in Estonian schools and the university (Fig. 5). The larger portion of students had never participated (34%-39%) or participated only on a few occasions (39-48%) in a class with the flipped classroom approach. Only 4-5 percent of the students had participated often in such classes. However, it seems that students like the flipped classroom approach (right part of Fig. 3). 55% of the students liked or strongly liked flipped classroom while only 10% did not like it.

The survey included 25 statements about the flipped classroom approach (Fig. 6 and Fig. 7). The students' attitude to flipped classroom was even more positive than to the work in pairs. Around two thirds of the students indicated that the flipped classroom approach helps them to learn this course better (statement 1) and more (statement 2) than they would learn in a traditional class. The students like to work out the material before class (statement 3). Some of them wrote in the comments that they do not necessarily like this, because it takes time, but they understand that this is very useful. Others wrote that they like to work out the material before class, because they feel more confident coming to the workshop. The students also like that, as material is known and questions emerge, they can ask them during the workshop (statement 23). They are not afraid at all to work out the material before the workshop alone without instructor (statement 9).

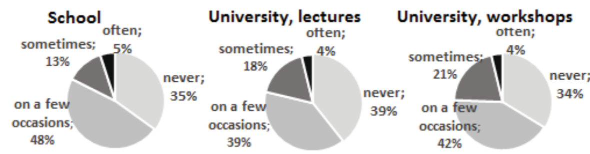



Figure 5. Student participation in flipped classroom in school and in university in lectures and in workshops

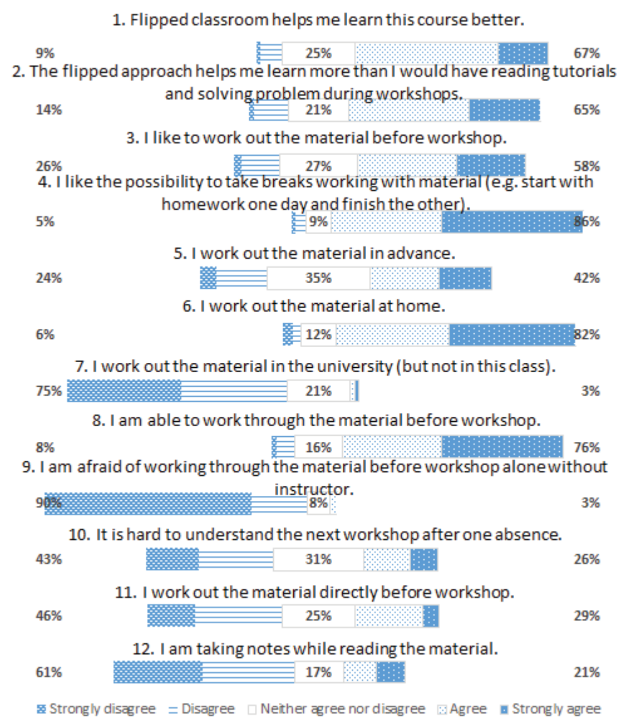



Figure 6. Likert responses to items connected to flipped classroom

The students indicated that they do homework at home (statement 6) and not in the university (statement 7). The students like the possibility to take breaks while working with the material (statement 4). However, the students do not take notes while reading the material (statement 12) (some of them commented that notes would be useful but they do not have time) and do not read several times (statement 13). The students are able to work out the material before the workshop (statement 8) and their homework is completely ready before the workshop (statement 16). The responses to the questions on working out the material in advance (statement 5) and directly before workshop (statement 11) were quite varied. There is also a large variance in the responses about understanding the material of the next workshop after one absence (statement 10). Some students wrote in the comments that they can say nothing about this because they did not miss any workshops.

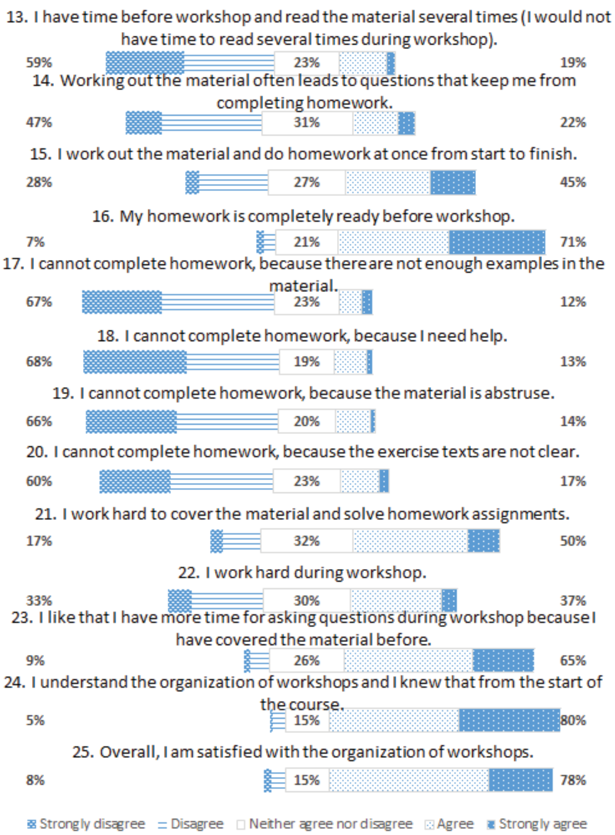



Figure 7. Likert responses to items connected to flipped classroom, part II

The material prepared for the students to work out before the workshop was rated as suitable. The material did not arouse many questions and the students could finish the homework (statement 14). The material has enough examples (statement 17) and is not abstruse (statement 19). The texts of exercises are clear enough (statement 20) and the students do not need help to complete homework (statement 18). In the comments, some students noted that if they needed help they consulted their friends or Google.

The students work harder before than during the workshop (statements 21 and 22). The students understand the organization of the workshop and they knew it from the beginning of the course (statement 24). Overall, the students seem to be quite satisfied with such organization of the workshops (statement 25). 55% of the students chose “Agree” as response to this question and 23% of the students strongly agreed with this statement.

We have seven instructors for workshops in 10 groups. The organization of the workshops was agreed before the beginning of the course. After 7 weeks, the instructors were asked for their opinion on the organization of the workshops. Six instructors indicated that they like that the students do work in pairs and fill in the worksheets, they supposed that the work in pairs makes the students work harder with homework. All instructors thought that flipped classroom is a good and reasonable approach and they liked it. The instructors liked that the students can work out the material at their own pace and are on the same level in the class. The instructors also noticed that the students are more active during the workshop. All instructors agreed that they did not want to change the flipped classroom approach to traditional class.

# 5 Conclusion

More and more educators start to use student-centered active learning approaches to support student learning. In this paper, we described the organization of workshops in the object-oriented programming course at the University of Tartu. We have integrated the flipped classroom approach into workshops so that the students read the prepared text material and solve necessary exercises before the workshop. During the workshop we use a collaborative approach: the students form work pairs during the workshop. We also brought peer review into the workshops. The paired students check each other’s homework using the prepared worksheet.

The students were asked about their attitude to the organization of the workshops, in particular the work in pairs and the flipped classroom. The survey with questions was used as a tool. Student response to work in pairs and the flipped classroom approach was largely positive. 78% of the students were satisfied with the organization of workshops.

There were two research questions of interest. What do the students think about the work in pairs in the programming workshop? What do the students think about flipped classroom in the workshop? Half of the students think that they get more knowledge and skills thanks to the work in pairs, they get advice and explanations from their peer and the outcome of the work becomes better. Two thirds of the students think that the flipped classroom approach helps them to learn better and more than they would learn in traditional class. Since students approved the organization of workshops, we will continue with that in the coming years. The text comments provided by the students are very useful and will be taken in consideration for improving the course.

However, it should be noticed that the results of the study do not provide a comment on the success (or otherwise) of the students in learning to program. Further research could be required to investigate that. In addition, the attitude of students from different workshop groups with different instructors can be investigated. If an instructor does not like the work in pairs, it is possible that the students feel the same.

This research was supported by the European Union through the European Regional Development Fund.

# 6 References


[1] L. Abeysekera, P. Dawson. "Motivation and cognitive load in the flipped classroom: definition, rationale and a call for research". Higher Education Research & Development, vol. 34(1), p. 1-14, 2015.

[2] T.M. Andrews, M.J. Leonard, C.A. Colgrove, S.T. Kalinowski. "Active learning not associated with student learning in a random sample of college biology courses". Life Sciences Education, vol. 10(4), p. 394-405, 2011.

[3] J.W. Baker. "The 'classroom flip': Using web course management tools to become the guide by the side". 11th International Conference on College Teaching and Learning, 2000.

[4] M. Barg, A. Fekete, T. Greening, O. Hollands, J. Kay, J. H. Kingston. "Problem-Based Learning for Foundation Computer Science Courses". Computer Science Education, vol. 10(2), p. 109-128, 2000.

[5] J.L. Bishop, M.A. Verleger. "The Flipped Classroom: A Survey of the Research". 120th American Society of Engineering Education Annual Conference & Exposition, 2013.

[6] E. Bojinova, J. Oigara. "Teaching and Learning with Clickers in Higher Education". International Journal of Teaching and Learning in Higher Education, vol. 25(2), p. 154-165, 2013.

[7] A. Butt. "Student views on the use of a flipped classroom approach: evidence from Australia". Business Education & Accreditation, vol. 6(1), p. 33-43, 2014.

[8] L.J. Cronbach. "Coefficient alpha and the internal structure of tests". Psychometrika, vol. 16(3), p. 297-334, 1951.

[9] G.C. Gannod, J.E. Burge, M.T. Helmick. "Using the inverted classroom to teach software engineering". Proceedings of the 30th International Conference on Software Engineering, p. 777-786, 2008.

[10] M. Guzdial. "Why is it so hard to learn to program? ". In Andy Oram and Greg Wilson, editors, Making Software: What Really Works, and Why We Believe It, O'Reilly Media, p. 111-124, 2010.

[11] M. Lage, G. Platt, M. Treglia. "Inverting the classroom: A gateway to creating an inclusive learning environment". Journal of Economic Education, vol. 31(1), p. 30-43, 2000.

[12] V. Leping, M. Lepp, M. Niitsoo, E. Tõnisson, V. Vene, A. Villems. "Python Prevails". Proceedings of the International Conference on Computer Systems and Technologies: CompSysTech'09, p. IV.17-1 - IV.17-5, 2009.

[13] J. Michael. "Where's the evidence that active learning works?". Advances in Physiology Education, vol. 30(4), p. 159-167, 2006.

[14] J. O' Kelly, S. Bergin, S. Dunne, P. Gaughran, J. Ghent, A. Mooney. "An Overview of the Integration of Problem Based Learning into an existing Computer Science Programming Module". Present at Pleasure By Learning conference, 2004.

[15] M. Prince. "Does Active Learning Work? A Review of the Research". Journal of Engineering Education, vol. 93(3), p. 223-231, 2004.

[16] J.C. Rodríguez-del-Pino, E. Rubio-Royo, Z.J. Hernández-Figueroa. "A Virtual Programming Lab for Moodle with automatic assessment and anti-plagiarism features". Proceedings of the 2012 International Conference on e-Learning, e-Business, Enterprise Information Systems, & e-Government, 2012.

[17] J. Sitthiworachart, M. Joy. "Computer support of effective peer assessment in an undergraduate programming class". Journal of Computer Assisted Learning, vol. 24(3), p. 217-231, 2008.

[18] H. Søndergaard, R.A. Mulder. "Collaborative learning through formative peer review: pedagogy, programs and potential". Computer Science Education, vol. 22(4), p. 343-367, 2012.

[19] J. Strayer. "How learning in an inverted classroom influences cooperation, innovation and task orientation". Learning Environments Research, vol. 15(2), p. 171-193, 2012.

[20] K. Topping. "Peer assessment between students in colleges and universities". Review of Educational Research, vol. 68(3), p. 249-276, 1998.

[21] S. Turner, M.A. Pérez-Quiñones, J. Chase. "Peer review in CS2: conceptual learning". SIGCSE '10 Proceedings of the 41st ACM technical symposium on computer science education, p. 331-335, 2009.

[22] X. Zhang, C. Zhang, T.F. Stafford, P. Zhang. "Teaching Introductory Programming to IS Students: The Impact of Teaching Approaches on Learning Performance". Journal of Information Systems Education, vol. 24(2), p. 147-155, 2013.

[23] Y. Wang, H. Li, Y. Feng, Y. Jiang, Y. Liu. "Assessment of programming language learning based on peer code review model: Implementation and experience report". Computers & Education, vol. 59, p. 412-422, 2012.

# Improving a Computer Science 1 Course with Flipped Instruction and Faculty Guides

**J. Hollingsworth and D. Powell**
Department of Computing Sciences, Elon University, Elon, NC, USA

**Abstract -** *This paper describes and discusses the advantages and disadvantages, for both students and faculty, of flipping an introductory Computer Science course by leveraging an already existing MOOC. Course content and format are described along with the new role of the faculty member (Guide) for this emerging course style. We argue that if faculty members embrace their new roles as Guides, a flipped Computer Science 1 course can be both compelling and rewarding for undergraduates.*

**Keywords:** Online courses, flipping the classroom

## 1 Introduction

MOOCs have changed and are changing the learning landscape in Computer Science. High quality, free or low cost, interactive, video based courses are available from a wide range of vendors and cover many aspects of the undergraduate Computer Science curriculum. The hypothesis of this paper is that an entry level Computer Science 1 course can be as effective or more effectively taught using flipped instruction, leveraging professionally developed content from third parties, than the traditional lecture format. For the past three years at Elon University the introductory Computer Science course called Computer Science 1, has been traditionally taught using a course text, Big Java Early Objects, with classes meeting 3 times a week in a computer lab setting [1]. Each class session typically involved a lecture followed by hands-on individual or group exercises. Faculty and students are very comfortable and successful with this model. In the spring 2015 semester the course was flipped with students out-of-class activities requiring the viewing of assigned interactive videos and in-class time spent on individual and group coding activities [2]. Faculty roles changed from lecturing the student to guiding the student. Faculty spend class time answering individual questions or leading problem solving exercises that demonstrated a programming concept or best practice. This paper describes the format and content of the course, the advantages and benefits to students and faculty, the disadvantages and the lessons learned.

## 2 Course Content and Format

During the spring 2014 semester, the Elon Computing Sciences Department gained experience in flipping an elective course on Advanced Programming for juniors and seniors majoring in Computer Science by curating a course with content from CodeSchool, Lynda and Udemy [3]. The experience gained here was used in developing the format and content of the Computer Science 1 course, in selecting a high quality 3rd party MOOC vendor, in selecting and developing course materials, in planning the 38 semester class session activities and in developing an individual assessment plan. Unlike the spring 2014 class made up of upper class Computer Science majors, this class was made up of freshman from a wide variety of majors including: Communications, Music, English, Business, Math, Engineering and Computer Science. Various high quality MOOC vendors such as Code School, Coursera, Lynda, Udacity and Udemy were searched for a beginner introduction to Computer Science course using Java and covering programming concepts using an objects-first approach. An Udacity course called Introduction to Java Programming was selected. This course is taught by Cay Horstmann, a professor at San Jose University, who is the author of the textbook, Big Java Early Objects, which is used in our traditional course [4]. This online course was an ideal one to choose for the transition to a flipped classroom as it followed the content of the text in sequence. The course is well designed with a large number of short, typically 1–3 minutes, high quality, professionally created videos with interspersed interactive quizzes arranged in lessons. The course also included problem sets that require Java code to be developed and submitted for online grading, forums for asking and researching already asked and answered questions and a dashboard to track completion status. The course has free course materials and has been well vetted and taken by over 200,000 people.

Table 1 shows the Udacity lessons mapped to chapters in the Big Java Early Objects book. The mapping is identical and greatly facilitated the transition from a traditional lecture based course to the flipped classroom using the Udacity materials. The syllabus is almost identical in the amount of time used to cover each topic. In fact, the Udacity materials are much more engaging and require more interaction then the static text book pages. For example, Table 1 shows that 27 videos cover the content in Book Chapter 1. These videos are all short, ranging from 14 seconds to 151 seconds, in order to keep the student fresh and actively engaged. The videos are interspersed with required fill in the blank or coding challenges that must be completed for the student to get credit for it on the course dashboard. The challenges typically require user interaction in one of every two videos. Of the 27 videos comprising lesson one 17 (63%) required a correct user response or code snippet. The code snippets are submitted through a browser window. Test cases are automatically run at Udacity and then the success or failure results are displayed in the browser. If the user cannot solve the problem then the user can choose to continue and the solution is presented and described in the next video. The lessons match the associated book chapter in an identical sequence and are by and large appropriately chunked to allow a student to complete each lesson in a two-hour period before class. A good example of the chunking is Chapter 7 on Arrays and ArrayLists. The lesson is broken into three pieces with lesson 7-1 covering ArrayLists in 44 minutes of videos, lesson 7-2 covering one-dimensional arrays in 25 minutes and lesson 7-3 covering two-dimensional arrays in 22 minutes. The proper size chunking greatly facilitates the development of the course syllabus and the out of class assignments.

As shown in Table 1, many of the lessons are followed by a problem set generally consisting of 20 shorts problems requiring input of one to four lines of code and 3 problems requiring a more substantial amount of code to solve. The user must correctly submit the code to get credit on the dashboard. The solution code is not directly provided after each problem as in the lessons. The problem sets are ideal to assign as in class activities for students to work on during a class session so they can discuss any problems with nearby classmates and the instructor. Moreover, problems sets provide a motivational tool to allow students who readily understand the material to complete it before class instead of during class and be rewarded with optional class attendance.

The use of lessons and problem sets with the direct mapping to book chapters made the transition for a traditional class easy. Our course and learning outcomes are unchanged. The same number of classes was used to cover each chapter but instead of spending time during class lecturing and classes exercises being done outside of class the student were solving problem sets and class exercises during class and watching and interacting with the videos for the lesson outside of class. In both the traditional and flipped classroom, each chapter concluded with an assessment.

In order to provide additional motivation to the students to complete the video assignments before class and the in-class problem sets before the next class, students were required to submit to the course management system (Moodle) a screen-capture of their dashboard showing the completion status of each video interactive exercise in a lesson and each problem in a problem set. The student received a score of 0 for incomplete, 1 for 25% correct, 2 for 50% correct, 3 for 75% correct and 4 for completely correct. The total weight of the snapshot completions to the course grade was 37.5%. The high percentage proved very motivating with 95% of all students completing on time all 11 lesson chunks and 8 problem sets with over 75% complete on each.

The students were still required to purchase the book Big Java Early Objects. The book was primarily used as a reference and as an easy way to review the concepts from the chapter covered in the Udacity videos. Though the interactive videos are a great way to learn, they are difficult to use as a quick and efficient review for a quiz.

**Table 1: Udacity Flipped Classroom Materials Mapped to Book Chapters Used in Traditional Course**

| Topic | Lesson | #Videos | Number and % Interactive | Total Time Minutes | Shortest Seconds | Longest Seconds | Problem Set | Book Chapter |
|---|---|---|---|---|---|---|---|---|
| Intro | 1 | 27 | 17 – 63% | 41 | 14 | 151 | 0 | 1 |
| Objects | 2 | 56 | 27 – 48% | 55 | 15 | 222 | 1 | 2 |
| Classes | 3 | 45 | 26 – 58% | 57 | 10 | 173 | 2 | 3 |
| Data Types | 4 | 50 | 35 – 70% | 80 | 6 | 224 | | 4 |
| Decisions | 5.1 | 13 | 9 – 70% | 23 | 19 | 41 | 3 | 5 |
| | 5.2 | 13 | 11 – 85% | 14 | 4 | 200 | | |
| Loops | 6.1 | 34 | 23 – 68% | 42 | 5 | 128 | 4 | 6 |
| | 6.2 | 40 | 35 – 87.5% | 67 | 8 | 217 | 5 | |
| ArrayLists and Arrays | 7.1 | 28 | 20 – 71% | 44 | 6 | 168 | 6 | 7 |
| | 7.2 | 19 | 11 – 58% | 25 | 6 | 143 | 7 | |
| | 7.3 | 11 | 6 – 55% | 22 | 23 | 139 | | |
| Discovering Classes | 8 | 27 | 22 – 81% | 59 | 17 | 392 | 8 | 8 |

# 3 Assessment

Since the majority of class time is spent collaboratively working on problem sets and class exercises, the focus on the chapter assessment in the flipped classroom is on the individual mastery of the chapter content and being able to demonstrate the ability to apply it to solve a problem. An entire class is spent on each chapter for individual assessment. Content knowledge is tested through a 20 question multiple-choice test and individual coding assignment containing one or more problems to code using the BlueJ IDE. The students receive immediate feedback on the multiple-choice portion of the quiz. The individual coding assignment is graded and returned by the next class. This approach differs slightly from the traditional classroom coding portion which was typically done outside the classroom and in pairs.

Though the grading of hands on coding has changed from pair homeworks to the in class individual coding problems, the quiz questions were kept the same as the previous traditional course for Chapters 1 – 4 and 6 – 8. Note: Chapter 5 in the traditional course did not have a multiple-choice quiz but used an alternative assessment mechanism. The intent was to use identical quizzes to provide comparative data on concept mastery between the flipped and the traditional model. Table 2 shows the average quiz grades of 29 students who took Computer Science 1 with the traditional approach in spring 2013 and 27 students who took Computer Science 1 with the flipped classroom in spring 2015. Both classes had the same instructor and the same book. Though only a comparison with one class, the data indicates that the flipped classroom results are equal to those of the traditional classroom. Since the resulting outcomes are similar or slightly improved, let us consider the advantages or benefits and disadvantages of the flipped classroom for students and faculty.

# 4 Student Advantages and Benefits of the Flipped Classroom

The spring 2015 flipping of the Computer Science 1 course provided students the following benefits and advantages:

- The interactive videos were available 24x7 and could be accessed from any location with an Internet connection. Students were able to control the learning environment and the time so as to be optimal for their individual need. There was no missed instruction due to an illness, a job interview, or conflict with other activities or classes. Students replayed as often as necessary parts of a lecture that proved difficult to grasp or were missed due to an outside distraction.
- Each instructional interactive video was professionally done and of high quality. Students could control the size of the screen, the screen resolution, the volume of the video and the lighting in their room to be optimal for them. This is in sharp contrast to the less controlled and perhaps not ideal lighting, background noise, instructor's penmanship, data projector sharpness and seating location in a classroom.
- All instructional materials were well vetted. Any mistakes or unclear aspects in the interactive videos had been mentioned in the Udacity forum and corrected. In addition, the course instructor pointed out mistakes or difficult problems in upcoming assignments.
- Morale was improved during class time. Coming to class provided the students with social interaction and the ability to work together to solve exercises instead of passively listening to a lecture. The benefits of pairwise programming can be more readily achieved as the instructor can observe and comment on pairwise programming techniques. Similar to using pairwise programming in the workplace during work hours, there is a dedicated class time for students to work with their partners. This overcomes the challenge and obstacles of scheduling time outside of class [5].
- Flipped instruction is preparing students for lifelong learning. The only constant is change and the ability to continually learn and update skills is critical for short term and long term success.

**Table 2: Comparative Quiz Scores for Traditional 2013 Class and Flipped 2015 Class**

| Class | Quiz 1 | Quiz 2 | Quiz 3 | Quiz 4 | Quiz 6 | Quiz 7 | Quiz 8 | Average |
|---|---|---|---|---|---|---|---|---|
| 2013 | 72.44 | 78.66 | 72.57 | 80.57 | 72.71 | 59.71 | 70.64 | 75.02 |
| 2015 | 78.33 | 87.76 | 76.14 | 73.93 | 73,89 | 72.96 | 62.16 | 81.55 |

# 5 Faculty Advantages and Benefit of the Flipped Classroom

Changing to a flipped classroom and becoming a *Guide* was a new and rewarding experience for faculty. The majority of class time was spent working with individuals, guiding students to solutions and answering questions as opposed to lecturing. Since the interactive instructional lessons were professionally prepared, faculty had more time to prepare class exercises, grade quizzes and spend preparing supplemental materials. Faculty were able to more easily monitor and evaluate course progress with dashboards clearly showing student progress and understanding. Finally, the faculty teaching Computer Science II, downstream from the flipped classroom, knows that all students have had a similar experience.

# 6 Disadvantages to Students

In a well-designed and executed flipped Computer Science 1 classroom, there are minimal disadvantages to the student. The primary obstacle is making the transition from traditional lectures to a flipped approach. In a traditional class students are accustomed to taking notes and underlining key passages in the text to facilitate learning. In the flipped instruction, note taking is done differently and there is no highlighting of video material.

# 7 Disadvantages to Faculty

There are three possible disadvantages to faculty. First, faculty need to be *Guides* instead of lecturing experts. The changing from preparing a class and giving it as a lecture is a significantly different approach and requires some retraining or exposure to develop the new style. Second, the course offering and reliability is dependent on the vendor. Udacity could close down operation or change their pricing strategy from offering free course materials to requiring the $199 dollars for official course enrollment at any time. If this happens in the middle of the semester then the faculty member must be flexible enough to quickly transition to the traditional role. In this event, our requiring of the course text provides a nice fall back to smoothly move into this style. Finally, The use of a MOOC course with a supporting textbook is an ideal situation. In fact, the authors believe this approach will become more prevalent as current textbook authors update their new course materials to include a MOOC to remain competitive. However, for many courses there may not be a MOOC with a matching text book or a MOOC that matches the complete set of expected course outcomes or a MOOC that does not have an integrated set of student quizzes. In these cases, the MOOC is still very valuable but the instructor will need to spend a substantial amount of time to find and curate the materials and software for the course and smoothly blend them together for the optimal user experience.

# 8 Conclusions

Flipping the classroom by wrapping an industrial strength Udacity MOOC course vetted by students and professionals throughout the world with high quality, professionally made videos, interactive exercises, dashboards, and forums greatly improved the traditional PowerPoint, lecture/lab based Computer Science I course at Elon. Instead of the majority of the 38 semester courses sessions being filled with 30 minutes or more of lecture, the time was spent by students applying computer concepts to solve faculty guided class exercises. Most faculty do not have the time or resources to create the high quality interactive, web based educational materials created by vendors which serve a large customer base. Why not leverage these materials in a flipped classroom and become a faculty *Guide* to more efficiently and effectively use one's computer science expertise to guide and individually help students in the local classroom lab setting? Though the data provided here only compares one Computer Science 1 course taught using flipped instruction against a single traditionally formatted course, it leads us to believe that we can achieve improved student performance with the flipped format. Elon plans to move to the flipped classroom format for Computer Science 1 in the upcoming academic year.

# 9 References


[1] Horstmann, Cay. "Big Java : Early Objects". 5th Edition. Wiley. 2013.

[2] Brame, Cynthia. "Flipping the Classroom". Avallable: http://cft.vanderbilt.edu/guides-sub-pages/flipping-the-classroom/ .

[3] Powell, D. and Hollingsworth, J. "Flipping the Classroom by Using Cloud Services and Third Party Online Courses in Traditional Class Settings Enables All Computer Science Students and Equal Education", Proceedings of the 2014 International Conference on Frontiers in Education, Computer Science and Computer Education. 2014. .

[4] Horstmann, Cay, Lee Cheng-Han and Tansey, Sara. " Intro to Java Programming". Udacity.com 2014. Available: https://www.udacity.com/course/intro-to-java-programming--cs046.

[5] Williams, Laurie and Kessler, R., All I Really Need to Know about Pair Programming I Learned in Kindergarten, Communications of the ACM, Vol.43., pp. 108-114. May 2000.

# SESSION

# RECRUITING STUDENTS TO COMPUTER SCIENCE, RETENTION METHODS AND STRATEGIES + ADDRESSING DIVERSITY ISSUES

## Chair(s)

### TBA

# A Public School Model of CS Education

**Leigh Ann DeLyser, Michael Preston**
NYC Foundation for CS Education, New York, NY, USA
leighann@csnyc.org, michael@csnyc.org

**Abstract—** *The New York City Foundation for Computer Science Education was launched in the summer of 2013 to ensure that all children in New York City public schools have access to computer science education. Over the past year and a half the foundation has worked with eight (8) programs to expand access to computer science, reaching more than 10,000 students in the 2014-2015 school year alone. In this paper we present a brief history of computer science education in K12 schools, focused primarily on New York City. We detail the philosophy of the foundation and highlight the programs we support and the impact they are having on New York City public school students. It is our hope that the unique approach of CSNYC in supporting research proven initiatives and connecting teachers in communities of practice will inspire other advocates of computer science education to adopt our successful model. At the end of the paper we present upcoming expansions for the 2015-2016 school year and discuss the research challenges and opportunities available.*

**Keywords:** Public School Education, Non-Profit, Computer Science Education

## 1. Introduction

In 2012 Code.org released their first video, highlighting computer science professionals and celebrities encouraging students to try computer science. Prompted by the increasing need for a technical workforce, and at the same time a declining number of computer science majors at 4 year institutions, the video, and subsequent media push for students to try an "hour of code" for computer science education week in December of 2012 has unleashed a nationwide movement to bring access to computer science to K-12 education[1].

Although the majority of career opportunities in STEM (Science, Technology, Engineering, and Mathematics) professions require some proficiency in computer science, an overwhelming majority of students did not have access to rigorous computer science in public schools. In an analysis of the 2013 AP Computer Science Exam, Barbara Ericson highlighted a series of statistics exposing the lack of access by many students and the inequity of course taking across the nation [2]. In New York State, which includes the numbers for New York City, only 68 black and 150 hispanic students attempted the AP CS exam, with pass rates of 33% and 35% respectively. These numbers are shocking considering almost 310,000 students are enrolled in high school in NYC alone, where 40.5% of the student population is hispanic and 27.7% of the population is black. Additionally, students also lacked access at the school level as 50% of APCS test takers in New York City came from 3 high schools, although there are over 400 high schools in the city.

Despite the challenges of lack of pathways into computer science for a majority of public school students, New York City has a large and rapidly expanding technology industry, represented not only by major technology companies such as Google and Microsoft, but also by a large financial sector that employs a significant portion of the technology workers in New York[3]. The technology industry in New York is constantly attempting to recruit and retain employees with appropriate background, and the need to increase the pool of applicants with appropriate prior preparation enjoys support from industry and policy makers alike, as demonstrated by the creation of the NYC Tech Talent Pipeline [4].

Together industry, policy makers, public education, and a force of volunteers have been working to expand access to computer science in New York City. Although there are many models of summer and after school programs, such as Girls Who Code, All Star Code, and others, the growth of computer science at the public school level in NYC has been unprecedented.

In this paper we discuss the work of the New York City Foundation for CS Education (CSNYC) that has spearheaded many of the efforts to increase access to computer science in the public schools in New York City. Since its launch in 2013, CSNYC has supported computer science programs in public schools, including a significant amount of teacher professional development. Over the past year and a half, CSNYC has expanded its programs to over 110 schools, serving more than 10,000 students in the 2014-15 school year alone. We present the work of CSNYC as a model for other states or regions who work to grow computer science offerings in their area.

## 2. History

Computer science education in schools has seen a resurgence after a drop off from the 1980s. There are echoes of similarity to the earlier movement in today's implementations of computer science, but there are also a series of differences between prior attempts to infuse computing into public schools and current initiatives.

### 2.1 Early CS Education in Schools

In 1980 Seymour Papert published his seminal work, *Mindsorms: Children, Computers and Powerful Ideas*. The book focused on micro worlds, and the educational benefits of young children learning to control actors in those worlds through computer programming. Especially compelling were Papert's claims that engaging with micro worlds and computer programming produced rich transfer to other problem solving domains and increased students' academic performance across all subjects [5]. Papert initially worked with students from a variety of academic and socioeconomic backgrounds and the documentation of his research lead to confidence that schools everywhere could implement meaningful curriculum regardless of their student populations.

At that time schools invested in bringing computing to the classroom to support constructionism, the inquiry-based and project-based approach to learning that included programming micro worlds as a key component. At the same time a number of education and psychology researchers delved into Papert's claims of transfer from computer programming to other domains. Although many experiments were carefully designed and executed [6], few yielded results of the transfer claims made by Papert [7].

In the 1990s and 2000s with the rise of the internet, computer usage in schools refocused from programming to information seeking, research, and as vehicles for content delivery with educational software focused on more traditional disciplines such as mathematics or reading. Fewer students were given the option of taking computer programming, and many schools dropped the courses all together, resulting in the statistic from Code.org, estimating that only 1 out of every 10 high schools in the US offers a programming course in 2012[1].

### 2.2 History of CS Education in NYC

New York City was one of Papert's original sites, however went through many of the changes described in the previous section. Although work addressing and categorizing the digital divide in the early 2000s focused on simple access to computers at work and school, one can extrapolate the access to computer science education mirrored, if not was exaggerated by, similar factors[8]. Additionally, as the internet became the standard of accessibility and classroom practice, New York worked through several iterations of providing internet access to schools.

The focus on hardware, software, and internet access combined with the increased experience of the general population with general computing skills lead to the decline of professional development opportunities for teachers around computing and computer science education. New York State previously required completion of technology standards, most often implemented in a computer applications course, emphasizing the usage of computers by students. Today, those technology standards are no longer required for a standard New York State diploma, and fewer schools require even basic technology classes for all students.

Although there are over 400 high schools in New York City, a select few are able to admit students in a competitive fashion. For example, Stuyvesant High School, Bronx Science, and Brooklyn Tech all require students to apply and admit students based upon their grades, attendance, and performance on a selective exam. In 2012 50% of the APCS test takers in NYC came from these 3 schools, although these schools only represent 3.7% of the overall NYC high school population. A symptom of a larger problem, computer science had been seen as an extraordinarily difficult subject, appropriate only for the most gifted students. This belief lead to the majority of computer science offerings existing at selective high schools, relegating students in non-selective schools to no technology courses, computer applications courses, or potentially hardware support and repair courses focused on certifications such as A+ and Cisco.

The current climate in New York City, however, is much changed. Many of the programs described in the following sections specifically work with some of the most disadvantaged students in public schools and are having resounding successes.

## 3. Programs

The NYC Foundation for CS Education is a non-profit organization launched in the summer of 2013 to ensure that all children in the NYC public schools have access to computer science education that will put them on a pathway to academic success and a 21st century career[9]. Since 2013, CSNYC has provided support to 8 programs reaching over 10,000 students in 110 different NYC public schools. CSNYC believes that NYC is not a "one-size fits all" city and by providing schools with choice among various programs we offer the opportunity for principals and teachers to select programs that fit with school culture and student populations.

Different from many other organizations and businesses focused on providing access to computer science, CSNYC does not have a single endorsed curriculum, instead the foundation has supported local efforts such as the Academies for Software Engineering, the Software Engineering Pilot Program, and ScriptEd to develop rigorous CS curriculum tailored to NYC public school students and their needs. Additionally, CSNYC has identified research-proven computer science programs that match Department of Education priorities and facilitated the expansion of those programs in the NYC public schools. Programs such as Bootstrap, Scalable Game Design, Exploring Computer Science, and TEALS have partnered with CSNYC for assistance connecting with existing schools for recruitment, space and costs of teacher professional development, and ongoing support in NYC for the teachers teaching the program content. Finally, CSNYC believes that an important client of our services are the teachers, and we provide two monthly teacher meet ups where teachers can learn best practices, career options, and about our programs.

In the following subsections we detail briefly each program to provide a sense of the scope of CSNYC activities and areas of focus.

## 3.1 Academies for Software Engineering

The NYC Foundation for CS Education provides minimal financial support to two whole-school model computer science focused high schools. The Academy for Software Engineering (AFSE) and the Bronx Academy for Software Engineering (BASE) are two new high schools where every student takes computer science each year of high school. AFSE and BASE are non-selective public high schools. Students are matched with the school based on an algorithm that uses randomization to ensure all students have an equal chance at being placed with the school[10].

Table 1: AFSE and BASE Statistics

| | AFSE | BASE |
|---|---|---|
| Total Population | 345 | 225 |
| % Free Lunch | 71% | 90% |
| Demographics | | |
| Hispanic | 45.5 % | 60.9 % |
| Black | 28.7% | 28.4% |
| Asian | 9.6% | 7.6% |
| White | 9.9% | 2.7% |
| Gender | M: 79.4% | M: 84.9 % |

Students at AFSE and BASE are representative of the larger NYC student population in terms of ethnicity, socioeconomic status, and prior academic performance. Table 1 provides basic statistics for each school. AFSE is in its third year of implementation, and therefore has a 9th, 10th, and 11th grade (adding a new 9th grade class each year), while BASE is in its second year of implementation with a 9th and 10th grade. At full size, both schools will have just under 500 students each.

Both AFSE and BASE are preparing for certification as a Career and Technical Education (CTE) High School. CTE schools still require students to meet the college readiness graduation requirements of a regular high school, but include programs for Work Based Learning(WBL) and require a structured sequence of courses that culminate in a technical assessment. At AFSE and BASE the Work Based Learning components include job shadowing and internships, field trips, mentoring, community service, and creative experiences. The addition of creative experiences to the standard WBL gives students credit for participating in activities where they design and build technical solutions to problems. Currently students have worked on projects for the Verizon App Challenge [11], hackathons[12], and after school or summer programs requiring development or prototyping of technological solutions.

The curriculum at AFSE and BASE focus on software design and development. The certifying exam being proposed by the CTE program is the Advanced Placement Computer Science A exam in Java. Although not ideal, the exam offers an external validation of the skills and knowledge students obtain during the multi-year CS sequence. In 9th grade students take an Exploring Computer Science aligned course that covers the fundamentals of programming in Scratch, simple web design in HTML and CSS, data analysis and spreadsheet functions, and a revisit of the fundamentals of programming in Python. Additional courses offered cover programming in Java (pre-AP), AP Computer Science A, Data Structures and Algorithms, Advanced Web Design/Javascript, 3D Printing and Modeling, and a CS Principles course that is part of the official CollegeBoard pilot program.

Both AFSE and BASE are primarily funded by the NYC Department of Education as public high schools. CSNYC provides small amounts of supplementary funding to enrich student experiences. Additionally, CSNYC funds Leigh Ann DeLyser as a CS consultant for both schools. Dr. DeLyser's role at the schools is to facilitate the CTE program creation and approval process, provide consultation on the computer science curriculum development, assist in mentoring the computer science teachers, and engaging with the school to help surface outside-of-class activities for students.

## 3.2 Software Engineering Pilot Program

The Software Engineering Pilot (SEP) Program was launched in the spring of 2013 with 20 schools, 10 high schools and 10 middle schools. After the first year the program now has 9 high schools and 9 middle schools after releasing two schools for lack of participation. Schools participating in the program identify two teachers who attend an intensive summer institute of professional development as well as once monthly sessions on the computer science concepts and pedagogy taught in SEP courses. In the 2014-2015 school year the SEP program provides ongoing professional development to 42 teachers and over 2700 students are enrolled in SEP courses.

The SEP program brings a 3 (middle school) or 4 (high school) year sequence of computer science and software engineering to existing public schools in New York City. Students in SEP classes study programming fundamentals in scratch, basic web design with HTML and CSS, mobile app development with Touch Develop, physical computing and robotics, and more advanced topics in programming with Python and Java. Currently in its second year of implementation the SEP program is working on expanding the curriculum offerings to include AP CS Principles and AP Computer Science A.

In addition to the classwork, students at SEP schools also engage in out-of-class learning experiences including field trips to companies with technical workers such as Etsy, Microsoft, and Bloomberg, and participation in hackathons. The SEP program puts on two student-focused hackathons per year, each hackathon centered around a theme and offered at multiple locations to provide proximity to schools in different boroughs around the city. Students in high school can participate in the SEP Pathfinders Job Shadow Program [13]. The Pathfinders Program provides stipends for students who work in companies in NYC and observe technical staff as well as use their skills to provide web design assistance, basic quality assurance, and office support. The Pathfinders Program is supported by a generous grant from the AT&T Foundation.

## 3.3 Bootstrap

Bootstrap is a curricular module for use in a mathematics course, or a stand alone curriculum for computer science that teaches algebraic and geometric concepts through computer programming[14]. The Bootstrap curriculum was created by Emmanuel Schanzer, building off the work of Matthias Felleisen and Program by Design. In the Bootstrap curriculum are detailed lessons, aligned to Common Core Mathematics Standards, with student handouts. During the module students design and program a 2 dimensional video game that can be played on any device and shared with friends.

The Bootstrap curriculum is closely aligned with mathematics content standards and throughout the module students solve word problems with the design recipe in order to add functionality to their game. The design recipe makes explicit the decomposition and understanding of a word problem by requiring that students write the name of the function they are writing, the domain and range of their function, and create examples (test cases) for the function. Throughout the curriculum students practice writing functions in Scheme, a functional programming language which is very similar to functional notation in mathematics, based upon their answers to the parts of the design recipe.

CSNYC supports the Bootstrap program by advertising professional development and assisting in school recruitment. Additionally, CSNYC pays participant costs for the professional development of NYC public school teachers for expansion of the program. With an initial school recruitment goal of 10 schools for 2014-15, CSNYC has helped facilitate the training of over 60 current and pre-service teachers and 23 schools have implemented the bootstrap curriculum module as of the writing of this paper.

## 3.4 Scalable Game Design

The Scalable Game Design program originated at the University of Colorado Boulder[15]. The Scalable Game Design project effectively leads students from creating games to programming advanced simulations. The project is the largest study of middle school computer science education in the United States with over 10,000 students from Alaska to Texas. In NYC, with the support of CSNYC for professional development and software costs, 16 teachers from 11 schools attended summer professional development and have implemented at least one Scalable Game Design module in grades 4-8.

A key component of the Scalable Game Design program is the use of Computational Thinking Patterns to define behaviors first in the game and then move on to simulations. The Computational Thinking Patterns have been shown to facilitate transfer between the game environment and simulation environment for behaviors such as collision. Additionally, the visual programming environment of AgentSheets and AgentCubes, the software used by the Scalable Game Design program, gives students a visual stage to conceptualize and debug their desired algorithms[16].

Additionally, teachers in NYC have extended the original set of modules and activities to enable students to create their own games stories and play, or to work on games related to topical themes. One school was completing a unit on westward expansion, and the teachers modified the original Frogger and Journey games in order create a game highlighting the different segments of Lois and Clark's journey from the book the class was reading.

## 3.5 Exploring CS, Code.org, and iZone

An important role for CSNYC to play in NYC is to help facilitate partnerships between organizations inside and outside New York. The iZone is a community of schools and partners who are committed to meet the individual needs of students through blended learning - using technological solutions to help provide differentiated instruction and resources. In partnership with the iZone, Code.org is implementing its high school curriculum. The first course offered in the school year 2014-2015 is Exploring Computer Science[17].

The Exploring Computer Science curriculum has proven successful with inner city students in over 8 years of research with the Los Angeles Unified School District. The curriculum takes a breadth based approach to computer science including units on Human Computer Interaction, Problem Solving, Web design, Programming, Computing and Data Analysis, and Robotics. The full curriculum is available from the Exploring Computer Science website at http://exploringcs.org.

In New York City, 36 schools (72 teachers) attended the summer professional development institute for Exploring CS and are implementing the program this school year. Future expansion is planned to include training more schools in Exploring CS, and expanding to Code.org's CS Principles curriculum for a second year in experienced schools. The Code.org and iZone partnership currently serve over 5,000 students.

## 3.6 ScriptEd

Scripted is unique in the CSNYC portfolio as it started with an after school program. ScriptEd has expanded to include in-school classes in addition to after school programs.

The ScriptEd model uses professional software engineer volunteers to deliver a curriculum focused on web design with HTML, CSS, and JavaScript and practical skills such as using GitHub and wire framing tools to produce products. Students are motivated to complete the ScriptEd program by the promise of internships using their newly acquired technical skills.

The ScriptEd partnership with CSNYC serves approximately 300 students in 15 schools in the 2014-15 school year. Students in the ScriptEd program are diverse both in ethnicity (30% Black, 43% Hispanic, 24% Asian, and 3% White) and gender (50% Male, 50% Female). Students completing the ScriptEd program have interned with companies such as American Express, JP Morgan, Contently, ThoughtWorks, About.com, and Getty Images among others.

### 3.7 TEALS

The TEALS (Technology Education and Literacy in Schools) program recruits, trains, mentors, and places technology professionals from industry in high school classrooms as volunteers. The volunteers work closely with school teachers throughout the program implementation in schools in order to not only teach the students in the class, but to instruct the classroom teachers as well. TEALS uses a fading model where over the course of several years the instruction shifts from the technology professional supported by the classroom teacher, to the classroom teacher with support from the technology professional. In New York City TEALS currently offers two classes for schools to choose from, an introductory course based upon the Beauty and Joy of Computing curriculum[18] and the Advanced Placement Computer Science A course.

In the 2014-2015 school year TEALS had programs in 131 schools serving more than 6600 students. In New York City, TEALS works with 14 schools. The TEALS program works with CSNYC to help facilitate school recruitment and ongoing teacher and volunteer professional development. TEALS also hosts a city-wide computer science opportunity fair to help students from all computer science programs in NYC understand the career options available to technical employees. The 2015 opportunities fair had over 1200 students and more than 75 companies, universities, and organizations participating.

### 3.8 Teacher Meetups

Although many CSNYC programs include ongoing teacher professional development, CSNYC believes bringing together teachers, professional volunteers, interested parties, and organizations is an important component for creating a community of practice around the discipline of teaching computer science. CSNYC directly organizes, supports, and provides facilitation for two teacher meet up events each month. The CSNYC Educator Meetup occurs on a weekday evening, and the CSNYC Scratch Educator Meetup happens on weekends to allow for longer engagement.

The Educator Meetup generally involves a program containing ample time for networking, combined with featured speakers around a topic of interest to the community. Past topics have included encouraging more women in technology, the "Educators Guide to Careers in Computing", the maker movement, and NYC Computer Science program offerings with representatives from over 27 organizations. Attendance at the evening meetup events varies, ranging from 49 to 221 depending upon the topic. The Scratch Educator Meetups are run un-conference style where educators propose session topics and form an agenda in real time. The Scratch Educator meet up had an average attendance of 35 participants in the 2014-15 school year showing a steady rise in attendance during that time.

## 4. Future Growth and Research Challenges

CSNYC is at the beginning of its mission to expand access to computer science for every student in New York City. Along the way we anticipate facing many challenges, some already defined, and asking important research questions. In this section I detail some of the challenges currently faced by our programs, and the research questions we currently seek to address. These questions are not meant to be exhaustive, merely a way to communicate our current research priorities as we continue to look for partners who would like to study our programs and the various forms of computer science education being implemented in the New York City public schools.

### 4.1 Current Program Challenges

Although each individual program is unique, there are shared challenges between the programs.

#### 4.1.1 Finding Qualified Teachers

Similar to the lack of qualified applicants for technical positions in companies, schools and programs are scrambling to find qualified teachers to teach computer science. Many programs have resorted to professional development for in-service teachers to instruct them in both the content and pedagogical content knowledge of the computer science curricula specific to the program's needs. This effort is costly and CSNYC is currently in discussions with local colleges and universities in order to help identify potential pipelines for teachers, shape the pre-service preparation programs to meet the needs of the New York City programs, and ensure that teachers graduating from those programs are appropriately credentialed to qualify to teach in public schools.

#### 4.1.2 Providing Students With Real World Experiences

A number of our programs use internships as a culminating activity or an integral part of the growth of the student through the program. A significant challenge, despite being in a large city, is the availability of companies willing to support high school students during an internship. Many of the internships students have been placed in have been highly successful, and all of the programs enjoy an almost perfect retention rate for companies who host interns. There is a misperception, however, in companies who have not hosted an intern, of what types of problems public school students from New York City will bring to their offices. We are working to change these perceptions by highlighting individual students and success stories, however the constant need to continue recruiting as the programs grow is a challenge.

### 4.2 Planned Future Growth

In addition to the growth in enrollment of current programs in New York, we are committed to supporting two additional initiatives in NYC for the 2015-16 school year. CSNYC is a partner on a National Science Foundation grant to bring the Beauty and Joy of Computing CS Principles course to New York City teachers. Over the four year grant, CSNYC will work with three additional organizations (Educational Development Center, the NYC Department of Education, and the University of California at Berkeley) to develop new curricular materials and train 100 teachers in New York City. The Beauty and Joy of Computing course is designed to prepare students for the new Advanced Placement Computer Science Principles exam.

CSNYC will also continue to work with Code.org in the 2015-2016 school year. Currently CSNYC supports the partnership between Code.org and the iZone for high school computer science. In the summer of 2015 Code.org will begin training elementary school teachers in the elementary curriculum for grades K-5 in New York City.

### 4.3 Research Questions

CSNYC recognizes the unique opportunity presented in the collaboration of public schools, non profit organizations, content providers, and researchers. To this end we are exploring multiple avenues for collaborative work with faculty at nearby (and some far) institutions. Driving many of the research proposals being drafted is the need to (1) identify core computer science content sequences that prepare students for success in school, college, and potentially career, (2) best pedagogical practices that enable a diverse population of students to learn the fundamentals of computer science regardless of academic background, and (3) generalizable models of best practices the emphasize both the importance of ongoing teacher professional development and also the reality of limited budgets. CSNYC welcomes new collaborations with institutions of high education in the support of programs through volunteers or space for professional development activities, and also collaborations focused on addressing research questions appropriate for NYC public schools and CSNYC programs.

## 5. Acknowledgments

The NYC Foundation for CS Education would like to thank our donors, board of directors, and partners who have provided guidance and support in many forms during our initial growth. We recognize the many volunteers that our programs depend upon to teach the next generation of technology workers and leaders, and the companies that generously donate space for professional development events, student field trips, and partnership meetings.

## References

[1] Code.org, “Code.org website”, 3 2015.
[2] Barbara Ericson and Mark Guzdial, “Measuring demographics and performance in computer science education at a nationwide scale using ap cs data”, in *Proceedings of the 45th ACM technical symposium on Computer science education*. ACM, 2014, pp. 217–222.
[3] LinkedIn, “Linkedin economic graph research: New york city”, February 2015.
[4] “The tech talent pipeline links underserved new yorkers to stem jobs”, December 2014.
[5] Seymour Papert, *Mindstorms: Children, computers, and powerful ideas*, Basic Books, Inc., 1980.
[6] Seymour Papert, *The children’s machine: Rethinking school in the age of the computer*, Basic Books, 1993.
[7] Sharon McCoy Carver, “Transfer of logo debugging skill: Analysis, instruction, and assessment.”, 1986.
[8] Jennifer S Light, “Rethinking the digital divide”, *Harvard educational review*, vol. 71, no. 4, pp. 709–734, 2001.
[9] New York City Foundation for CS Education, ”, March 2015.
[10] “How game theory helped improve new york city’s high school application process”, December 2014.
[11] Verizon Foundation, “Verizon innovative app challenge”, Web Page, March 2015.
[12] “Two nyc teens head to the fifth annual white house science fair with apps they developed”, March 2015.
[13] Software Engineering Pilot Program, “Pathfinders job shadow program”, Web Page, March 2015.
[14] Emmanuel Schanzer, Kathi Fisler, Shriram Krishnamurthi, and Matthias Felleisen, “Transferring skills at solving word problems from computing to algebra through bootstrap”, *Circles*, vol. 6, no. 4, pp. 5, 2015.
[15] Alexander Repenning and Andri Ioannidou, “Broadening participation through scalable game design”, in *ACM SIGCSE Bulletin*. ACM, 2008, vol. 40, pp. 305–309.
[16] Alexander Repenning, “Conversational programming: Exploring interactive program analysis”, in *Proceedings of the 2013 ACM international symposium on New ideas, new paradigms, and reflections on programming & software*. ACM, 2013, pp. 63–74.
[17] Joanna Goode and Jane Margolis, “Exploring computer science: A case study of school reform”, *ACM Transactions on Computing Education (TOCE)*, vol. 11, no. 2, pp. 12, 2011.
[18] Andrew McGettrick, Eric Roberts, Daniel D Garcia, and Chris Stevenson, “Rediscovering the passion, beauty, joy and awe: making computing fun again”, in *ACM SIGCSE Bulletin*. ACM, 2008, vol. 40, pp. 217–218.

# Bridging the Gap: A Virtual Pre-College Program for Computer Science Students

**Rachel Hallie Lomax, Lena Alston, A. Nicki Washington, Ph.D.**

Department of Computer Science, Howard University, Washington, D.C.

**Abstract -** *Despite current technological boom, literature supports that African-American students in computer science (CS) are suffering from high attrition rates. Of the technological graduates, many of them are not qualified to enter the job market in their fields of study.*

*As Phase II of an initiative already in place to improve undergraduate computer science performance at Historically Black Colleges and Universities (HBCUs), the Pre-Freshman Experience in Computer Science (PREFICS) aims to address one of the dividing factors that contributes to a student's success or failure in computer science: prior experience level.*

*Students with no prior exposure to CS will be encouraged to take the online course before entering their freshman year. Having familiarized themselves with basic concepts and practices in the Python programming language, these students will have caught up to the students fortunate enough to have CS experience before college.*

**Keywords**: Computer science education, pre-college, attrition reduction

## 1 Introduction

Though the field of technology continues to advance everyday, the number of graduating professionals qualified to enter the field remains low. Part of this problem comes from an inability not only to gain students but to retain them as well.

Students who begin college with little to no background in computer science have a more difficult time staying motivated than those with prior exposure. Through the creation of an online summer course, we hope to give these students a chance to build a foundation for computer science (CS) concepts by learning without the added pressure of receiving a failing grade looming over them.

The remainder of this paper is organized as follows. Section 2 describes what motivated the conception of the course. Section 3 details previous studies of related work. Section 4 outlines the content of the proposed work. Section 5 summarizes the work and suggests future endeavors.

## 2 Motivation

Historically Black Colleges and Universities (HBCUs) are now facing what is often referred to as the “Quiet Crisis” [5]. The Quiet Crisis describes the issue of the under-representation of minority students—most notably, African-American students—in science, technology, engineering, and mathematics (STEM) career fields, as well as the lack of graduating seniors who are considered qualified to fill the available positions. This under-representation exists in spite of the fact that we are currently in the midst of a technological boom, where jobs in STEM fields are quickly rising in demand.

As the Baby Boomer generation enters the retirement age, the number of vacant positions in technological fields will only continue to increase. Blinded by the luxuries that the current boom in innovation has afforded us, we as a country have failed to fully realize how great of a hole we are digging for ourselves. This heavy reliance on technology, and the way that it has been so smoothly integrated into our everyday lives, will lead to a massive aftershock when there are no longer enough qualified professionals to continue to provide these services.

One solution to the problem at hand involves looking into the untapped talent in under-represented minority groups and increasing the number of STEM degree-seeking students from these groups [8][13]. Although the combination of women and people of ethnic minority backgrounds make up nearly two-thirds of the United States population, they represent less than one third of the science and engineering workforce—a workforce composed of 70 percent white males.

Increasing diversity in the technology industry serves more benefits than just filling vacant positions. It also improves innovation. A wider variety of people of different demographics working together bring about ideas that may have been overlooked by the majority. This could result in the invention of an entirely new product, or simply having different people to test an existing one [10].

Research suggests that providing students with a non-intimidating, comfortable learning environment increases the chances of success for these underrepresented groups. When placed in a classroom setting with peers from the overrepresented majority, students can face what is popularly referred to as “stereotype threat.” Stereotype threat is a phrase meant to describe the phenomenon that occurs when a fully capable student performs below expectations when reminded that they are in the minority of the people performing a task. It is believed that this happens because the student suddenly feels as if their performance directly impacts how their entire group is perceived, causing them to crumble under pressure. In a large lecture hall, a student from one of these groups may hesitate more than others to speak out or to ask a question, with fear of being perceived as incompetent and representing their demographic group as so.

It is also recommended that faculty advisors emphasize the importance of a strong mathematical background to prospective students. Doing this, while also consciously constructing exams in a way that reasonably builds off of learned material from the course, will lower the perception amongst students that luck was a significant factor in their overall academic success, or failure [12]. When professors write exams in a way that assumes prior knowledge of material outside of the scope of the curriculum, the tests scores become less of a reflection of capability to understand the coursework. The test scores distribution is now skewed to favor those privileged enough to have had experience outside of the classroom.

# 3 Related Work

Researchers at the University of York have attested that providing both students and teachers with immediate feedback on performance helps to improve the overall performance of the student [9]. Through immediate feedback, not only does the student know where they are falling short, but the educator also knows where the focus needs to be shifted in instruction. Finding a way to provide immediate feedback emulates individualized learning in a way that is scalable. In a traditional classroom, the teacher might ask questions in front of the class, and the students who know the answer may raise their hands to answer it. This doesn't tell the teacher whether or not every student understood the problem, and leaves it up to the students who didn't understand to say something about it. However, for classrooms that integrate interactive learning technologies into this process, the teachers could potentially keep track of every individual student's performance on every individual question. This could even show an evolving graph of when and where the student finally begins to understand what it is they are doing wrong. In cases where each lecture question is self-paced—where students can continue to answer questions as long as they get the previous one right— students who understand the material can continue to progress without being slowed down by those who are struggling, and those who struggle can stop being left behind.

A study by Georgia Institute of Technology (Georgia Tech) followed students who entered the university as computer science majors and tracked the progress of those they refer to as the “stayers” (students who remained with the program until graduation) and the “leavers” (students who began in CS but changed their majors) [2]. The most notable difference between those who stayed and those who left had to do with their overall perception of CS prior to entering the classroom. The “stayers,” on average, initially displayed more confidence in their ability to succeed than the “leavers.” Most “stayers” had prior exposure to CS, enjoyed solving problems, and were more likely to actively participate in CS-related activities outside of the classroom. Nearly half of the “leavers,” on the other hand, picked CS because of the lucrative job opportunity. They were also more likely to characterize coders as being “asocial,” as a result of the time-consuming nature of programming. They were less likely to associate a computer science degree with anything beyond simply learning how to write code.

A study of students at the University of New Mexico found that students from under-represented groups were more likely than their peers from over-represented groups to perceive their computer science courses as difficult [11]. More than half of the minority students even reported that they had been told that it was only possible to pass courses after taking them more than once. This led to more students falling off track to complete their degree on time, and makes them more likely to leave the major altogether. This perception often existed even before the students started the class and led to them not seeking help because they believed the material to be too advanced to understand the first time.

Culturally responsive teaching methods are very important when educating students of minority serving institutions. Many studies have shown the underachievement of African-American and Hispanic students in the United States education system. One mistake that teachers may make when teaching students of different cultural backgrounds is treating them all the same. In attempt to avoid discrimination, educators can actually end up neglecting to acknowledge the individual’s learning and communication style adopted from their cultural norms. Without generalization, different cultural groups do tend to possess different strengths and types of intelligences. For example, African-Americans and Latinos have a greater tendency to learn better cooperatively and prefer to work together, while Anglo-American students, on the other hand, prefer to work alone, a practice almost directly associated with the computer science field. This is not to say that African-American students should be excused of independent work, but the skills may need to be gradually developed as new material is introduced. Howard Gardner categorized eight different types of intelligences to consider: logical-mathematical, linguistic, musical, spatial, bodily kinesthetic, interpersonal, naturalistic, and intrapersonal (1993) [7]. A student's social, political, or geographic circumstance may make them more likely to excel in a particular intelligence than another student of a different cultural background. The suggested solution is to use as many teaching modes as possible when teaching culturally responsively to accommodate even the outlier individuals within the diverse cultural groups and to refrain from making generalizations about all members of those groups.

Recent literature supports the idea that the concept of gamification does not necessarily imply playing a game but utilizes game design components to influence user behavior [4][6]. As more research is being done around the new buzzword, we reject the notion that money, gifts, and points are the single most effective means to motivate. Rather than using money, gifts, points, or other seemingly tangible forms of reward, self-efficacy, group identification, and social approval provide a more powerful form of incentive and yield longer-term participation. Context also proves to be a major factor contributing to what kinds of gamification are appropriate for a given instance. Condensing the idea of gamification into badges, levels, and leaderboards can actually result in damage to existing interest and engagement. The process requires much more thought into each unique experience and the values that those users are driven by. The solution to better gamification strategies is more research in design and full understanding of the context.

At Howard University, first years in computer science are expected to take an Introduction to Systems and Computer Science course (CS0) during their first semester. The problem that has been noted over the past few years is that students who take CS0 still exit the course without enough knowledge to succeed in Computer Science I (CSI) [1]. A recent effort to better prepare students for the next levels came in the form of a partnership with Google and the beginning of a peer-lead tutoring "boot camp" program. The Googler in Residence program (GIR) was meant to give students access to real-life software engineers of color at Google, as well as create a more cohesive curriculum for CS0. The boot camp was meant to provide an additional resource for students to receive help and encourage peer-learning development. The boot camp is run by upperclassmen, and provides help for classes most-often taken during a student's first two years of study. The results of these past initiatives were an increase in the number of students enrolled in CS0, as well as a rise in the percentage of students who matriculated on to CSI.

These results, however, presented a new problem. When evaluating the students entering CS0, there was a pretty significant overall performance gap between those who entered with prior experience in CS and those without it. For the former group, the course moved too slowly. For the latter, the course moved too fast. As the course progressed, the students with little to no background fell further and further behind. They feel discouraged to seek help when needed because of how well the students with prior knowledge are performing. Many of these students will either have to repeat the course or leave without enough preparation for the next level of coursework. In efforts to keep the slower students from falling too far behind, the faster ones must tradeoff the benefit of being challenged enough to go on to the next level as prepared as they could be.

# 4 Current Work

In an effort to bridge this performance gap between the students at both ends of the spectrum, we have designed a self-paced, online virtual classroom for the incoming first year students to take the summer before they begin their studies. The course is intended for students will no prior experience in computer science. The goal is to allow these students to catch up to their peers by providing them with the experience that other students have already received. The project will pilot at Howard University in the summer of 2015. The curriculum is designed to cover the basics of programming concepts and practice using the Python programming language. The course will be hosted on Moodle with an open discussion board where students can help one another when they get stuck. The lectures will be presented as videos with transcriptions available for those who prefer to read the content. Five lessons are divided into 4 sections each, estimated to be completed over a 5-week duration. Each of the sections will be accompanied by a set of multiple-choice questions to assess the student's understanding of the concepts covered. Every lesson will conclude with a cumulative programming challenge that puts together all that they have learned thus far. A detailed outline of the curriculum is provided in Table 1.

**Table 1. PREFICS Curriculum**

| Lesson | Sections | Learning Outcomes |
|---|---|---|
| 1. Intro to CS | 1. What is CS?<br>2. What is programming?<br>3. Variables<br>4. Review | 1. Students will gain knowledge of the different career paths offered by Howard CS.<br>2. Students will be introduced to programming, its various languages, and |

| | | |
|---|---|---|
| | | algorithms.<br>3. Students will begin to practice using variables and their data types.<br>4. This section summarizes the week and allows students to practice writing algorithms. |
| 2. Basic Concepts | 1. Functions<br>2. Conditionals<br>3. Debugging<br>4. Review | 1. Students will practice using parameters and returns of user and Python built-in functions.<br>2. Students will begin to understand control statements using if, elif, and else conditionals as well as Boolean operators.<br>3. Students will be able to practice problem solving by finding errors in code.<br>4. Students will be tested on their understanding of functions and conditional statements by debugging faulted code. |
| 3. Data Structures | 1. Lists<br>2. Tuples<br>3. Dictionaries<br>4. Review | 1. Students will begin indexing and modifying the most basic Python data structure, lists.<br>2. Students will learn to differentiate tuples from lists in Python.<br>3. Students will learn another common data structure in Python, dictionaries.<br>4. Students will review the difference between data structures and practice using each of them. |
| 4. Loops | 1. Introducing loops<br>2. While loops<br>3. For loops<br>4. Review | 1. Students will be introduced to different types of loops and how they vary indifferent languages.<br>2. Students will complete exercises<br>3. Students will practice the three types of for loops in Python.<br>4. Students will get a chance to write a code that requires looping, data structures, and knowledge of debugging techniques. |
| 5. Projects | 1. Robot Challenge<br>2. GPA Calculator<br>3. Image Manipulation<br>4. Image Filter | 1. Students will be challenged to complete a series of tasks navigating a robot through a maze. The challenges will require the use of conditionals and loops to complete.<br>2. Students will practice writing their own full programs, start to finish.<br>3. Students will gain an understanding of image representation and RBG codes to be used in the next section.<br>4. Students will gain a deep understanding of Python functions, conditionals, data structures, and looping and the debugging process. |

By introducing the concepts described in Table 1 in a self-paced environment where their performance cannot negatively impact their overall grade, we are encouraging them to form a basic understanding of the material and increasing the likelihood of success when they enter CS0.

# 5 Summary

Prior initiatives to improve the performance of undergraduate computer science students at HBCUs have revealed another factor that hinders the success of incoming students: the drastic performance gap between those with previous coding experience and those without. This virtual pre-college program aims to alleviate the issue of an appropriate class pace while expanding and improving the perceptions of computer science when students enter their first year in CS0.

We hope that by incorporating different types of information, video and text, that we can accommodate for different learning preferences. To validate for the students their understanding of the material, the assessment questions reflect the content and examples from the lessons. This is to prevent the perception that luck led them to the correct answer and build their confidence in the material. Because the course is self-paced, students will not feel limited nor pressured to speed up or slow down when learning the new, or familiar, material. Traditionally and historically, students of color tend to be a more social people, those who value communal engagement and learn more cooperatively. The message board allows the students to answer each other’s questions and work together to alleviate their roadblocks. This also helps them to feel stronger group identification with their future classmates. The final project encourages the students to share their work on Instagram, a photo-sharing social media application. This will promote peer approval of the student’s field of study by people outside of the classroom, and thus, improve the perception of CS as a discipline.

In future implementations of this course, we plan to integrate an auto-grader that can run and test the code for the students and give them feedback in regards to edge cases. The course will also be modified based on the overall performance of its first round of students in CS0 this coming fall. The changes will be made based on what they seemed to understand, versus what they still appeared to struggle with, as well as personalized feedback from the summer course participants.

## 6 Resources


[1] Biggers, M., Brauer, A., & Yilmaz, T. (2008). Student Perceptions of Computer Science: A Retention Study Comparing Graduating Seniors vs. CS Leavers. ACM SIGCSE Bulletin, 40(1), 402-406. Retrieved November 12, 2014, from ACM Digital Library.

[2] Cooper, S., Dann, W., & Pausch, R. (2003). Teaching Objects-first In Introductory Computer Science. ACM SIGCSE Bulletin, 35(1), 191-195. Retrieved November 19, 2014, from ACM Digital Library.

**[3]** Deterding, S. (2012). Gamification: Designing for Motivation. Social Mediators.

[4] Jackson, S.A. "The Quiet Crisis: Falling Short in Producing American Scientific and Technical Talent." Building Engineering and Science Talent.

[5] Lee, J., & Hammer, J. (2011). Gamification in Education: What, How, Why Bother?

[6] Morgan, H. (2010). Improving Schooling for Cultural Minorities: The Right Teaching Styles Can Make a Big Difference. Educational Horizons, 88(2), 114-120. Retrieved December 3, 2014, from ERIC: Institute of Educational Sciences.

[7] Prepare and Inspire: K-12 Science, Technology, Engineering, and Math (STEM) Education for America's Future. (2010). Report to the President.

[8] Sheard, M., & Chambers, B. (2011). Self-Paced Learning: Effective Technology-Supported Formative Assessment. Institute for Effective Education.

[9] The Engineer of 2020: Project Prospectus. (n.d.). IEEE Engineering Management Review.

[10] Varma, R. (2006). Making Computer Science Minority-Friendly. Communications of the ACM, 49(2), 129-134.

[11] Washington, A., Anderson, K., Warner, G., & Burge, L. (2010). Graduating Engineers in the U.S.: Bridging Engineering and Entrepreneurship for African-American High School Students.

[12] Washington, A., Burge, L., Mejias, M., Jean-Pierre, K., & Knox, Q. (n.d.). Improving Undergraduate Student Performance in Computer Science at Historically Black Colleges and Universities (HBCUs) through Industry Partnerships. *SIGCSE '15 Proceedings of the 46th ACM Technical Symposium on Computer Science Education*, 203-206.

[13] Wilson, B., & Shrock, S. (2001). Contributing to Success in an Introductory Computer Science Course: A Study of Twelve Factors. ACM SIGCSE Bulletin, 33(1), 184-188. Retrieved November 12, 2014, from ACM Digital Library.

# Retaining Women in Computing

**R. Davis**[1]**, and A. Arndt**[2]
[1]Computer Engineering, Santa Clara University, Santa Clara, CA, USA
[2]Extension Services Consultant, National Center for Women in Information Technology, Rising Sun, IN, USA

**Abstract -** *The number of women in the field of computing has been very low for decades. In order to determine which (if any) of the efforts we implemented to retain women has had a substantial impact on their decision to persist in the major, we surveyed all of the women who had been declared with a major in the Computer Engineering department for any length of time between 1992 and 2012. The survey asked the women to reflect on their preparation, expectations, experiences, motivation, and various influencers on their decision to persist or leave. We also tracked persistence in the field beyond graduation.*

*In this paper we describe the methodology used, provide a qualitative analysis of the results, and summarize the lessons learned that might guide possible changes at the institution to encourage more women to persist.*

**Keywords:** Retention, persistence, women in computing

## 1 Introduction

At Santa Clara University we have long been concerned about the gender diversity of the students studying computing. Over the years we have implemented a variety of initiatives to encourage more women to persist in the majors offered by the Computer Engineering department. We have increased our percentage of female students, although it is still well below where we would like to see it.

Historically, we have done well with retention at the university. If you look at graduation rates without paying attention to major at graduation, the women graduate at essentially the same rate (or slightly higher) as the men. The six-year graduation rate from SCU for both male and female students who enter the university as engineering majors is 86-89%. However, we do not keep as many women in the major. For example of those students who entered the major as freshmen in 1999 or 2000, 27% of the women students earned their Bachelor's degree in a major outside of engineering, while only 7% of the male students earned their degree in a major outside of engineering.

Realizing that we had a problem retaining women in the major, we implemented several initiatives aimed at improving retention.

We suspected that the most effective initiatives to retain women in computing were those that addressed the women's sense of belonging and being comfortable in the major. They need to feel welcome and that their contributions are significant, needed, and welcome. We believe that our women students want to make an impact on the world around them, and our job is to help them realize their potential to do so. While we believe every one of our initiatives was designed to encourage the students in this way, we needed to hear from the students what mattered most to them. Resources are always limited, and we want to spend them where they do the most good.

In order to evaluate the impact of the various initiatives as experienced by the students, we developed a survey to ask them what influenced their decisions to persist in or leave a major in the department of Computer Engineering.

## 2 Interventions Implemented

In 1999, we joined the inaugural group of universities (Texas A&M, MIT, Purdue, and SCU) in the Virtual Development Center, an initiative of Anita Borg's Institute for Women and Technology (now the Anita Borg Institute, ABI). Through the VDC we offered courses that worked with the local community to design projects that met specific needs. We found that these community-based projects were particularly attractive to female students (and other underrepresented groups in engineering). Our experience with community-based projects as a means to motivate and retain women and minority students has been reported on elsewhere [1,2,4,6].

In addition to the creation of community-based projects, we attempted to improve the self-efficacy of women students with a project designed to increase their identification as tech-savvy engineers. We offered female engineering students handheld computers (back in 1999-2000), and arranged for them to meet regularly to share their experiences and suggest improvements to the devices. This project was reported on at the ASEE conference in Honolulu and also at ICEER, Melbourne in 2007 [2,3].

In 2004, we joined the National Center for Women and Information Technology, committed to moving the needle for women's participation in computing. We enlisted with the NCWIT extension services to run a Student Experience of the Major survey in order to understand how our students viewed their learning environment. We also started the design of a new major in Web Design and Engineering, capitalizing on what we believed to be the most attractive aspects of computing to young women. While enrollment in this new major has been small, it has attracted a majority of female students from the very beginning. (Three of the first four students to declare the major were women, and today 65% of the Web Design majors are female.)

The Computer Engineering department joined the first cohort of NCWIT Pacesetters in 2009. As part of Pacesetters, senior executive leaders from twenty-four academic and corporate organizations put a stake in the ground to demonstrate that quantifiable change within their own organizations is possible. Collectively, we committed to adding one thousand “net new women” (women who would not have been there otherwise) to the technical talent pool by the end of 2011. The first Pacesetters cohort achieved that goal and more, adding 1685 women to the U.S. talent pool in just two years. We continued participation in this effort by joining the second Pacesetters cohort started in 2013, with a goal of adding another two thousand net new women.

At Santa Clara, we have initiated several programs, some aimed directly at women students in computing majors, but most are available to all engineering students.

- Mentoring activities
  - Sisters Program – each new student is paired with a continuing student to help her adjust to college life and engineering in particular.
  - Women in Engineering dinner – every fall term we have a dinner for all undergraduate women students and women faculty in engineering. (Our engineering faculty is over 33% female.)
  - Women in Computing lunch – at the end of every term, on the first day of exams, we provide an informal – come when you can, leave when you must – lunch for all of the women students and faculty in Computing.
- We fund women students attending the Grace Hopper Celebration of Women in Computing (GHC) and the Society of Women Engineers (SWE) conferences every fall.
- We send students to the Anita Borg Institute’s Women of Vision Awards Banquet every spring.
- We run several outreach activities:
  - One Step Ahead, a collaborative effort between our SWE student section and GAINS (Girls Achieving In Nontraditional Subjects–a program of the local high school district) brings high school girls to campus several times a year to learn about different fields of engineering.
  - SEEDs – Spring Engineering Education Days, a program introducing high school students to several areas of engineering on Saturdays in April. http://www.scu.edu/engineering/undergraduate/seeds.cfm
  - Summer Engineering Seminar – a residential program for underrepresented groups in engineering, introducing high school rising seniors to several fields of engineering. http://www.scu.edu/engineering/undergraduate/ses.cfm
  - GetSET – a collaboration with the local chapter of SWE that brings ~80 minority girls on campus for a residential weeklong introduction to several areas of engineering in the summer.
  - Many individual activities for various high schools and middle schools.
- Curriculum
  - We have added a pre-CS1 course for those who have never had any programming.
  - We created the Web Design and Engineering major.

These efforts have been successful, as we have increased the number and percentage of women students. However, we are still at only 24% women in all of engineering, and less than 18% in computing majors. In order to renew our efforts, focusing on those initiatives that have had the most impact on our women students, we designed a survey to ask the women what influenced their decisions to persist in or leave computing.

We suspected that the most effective initiatives to retain women in computing were those that addressed the women’s sense of belonging and being comfortable in the major. They need to feel welcome and that their contributions are significant, needed, and welcome. We believe that our women students want to make an impact on the world around them, and our job is to help them realize their potential to do so. While we may believe every one of our initiatives is designed to encourage the students in this way, we needed to hear from the students what mattered most to them. Resources are always limited, and we want to spend them where they do the most good.

## 3 Methodology

The study was conducted through an on-line survey of 42 questions using Survey Monkey, a cloud-based survey tool. We sent the survey to all women (we could find) who were declared as any undergraduate major in the department of Computer Engineering for any length of time between 1992 and 2012. In that time period, the department offered a Bachelor of Science in Computer Engineering (1992 – 2011), Bachelor of Science in Computer Science and Engineering (2009 – 2012), and a Bachelor of Science in Web Design and Engineering (2009 – 2012).

The degrees in Computer Engineering and in Computer Science and Engineering were the same in all but name. The degree actually started as a BS in Computer Science from a department of Electrical Engineering and Computer Science. When the department split into two departments, the chosen names for those departments were Electrical Engineering and Computer Engineering, simply because we exist in an Engineering School. The name of the computing degree was changed at that time to reflect the name of the department. We found that the name “Computer Engineering” led many prospective students to assume that it was a degree in computer hardware only, and felt that this name restricted our recruitment unnecessarily. “Computer Science and Engineering” has always been the most accurate description of the content of the degree, and in 2010 we went through the double effort to have it accredited by both the Engineering Accreditation Commission and the Computing Accreditation Commission of ABET.

The degree in Web Design and Engineering was specifically created to attract more female students. It requires a combination of computing, sociology, art, and communication courses that enables its graduates to develop the skills necessary to contribute to the world of new media and develop a broader perspective regarding the impact of computing on society.

A link to the survey was successfully emailed to three groups of former students:

- To 98 students who graduated between 1993 and 2005, or did not graduate, but were admitted between 1993 and 2000, of whom 26 responded;
- To 34 students who graduated between 2006 and 2012, of whom 14 responded;
- To 59 students who started between 2001 and 2012 and had not graduated by June 2012, of whom 27 responded.

We asked about the students' motivation coming into the program and their preparation for study in computing, as well as their perception of the impact of the program itself and various activities in which they participated on their persistence in computing.

# 4 Qualitative Analysis of Results

Findings are aggregated narratives of descriptive statistics for selected responses, with summaries, themes, and highlighted quotes derived from the open-ended items.

We looked at the intrinsic vs. extrinsic motivation to start the program and the students' prior experience with computing to see whether we should expect differences in completion rates. We compare the participation in community-building activities, such as clubs, conferences, and programs, of students who completed their degree in computing and those who left the field. We present the responses to questions regarding motivation, preparation, confidence, and involvement, disaggregated by whether or not the students persisted to a degree in computing.

## 4.1 Motivation to Major in Computing

We asked why the students initially decided to major in computing, and offered several possibilities they might select as well as the option to add their own comments. Table 1 indicates the students who selected each of the reasons presented, as a percentage of the students who completed a degree in computing, and as a percentage of the students who left computing. Those who left computing are further split according to whether they completed a degree in another discipline or they left the university without a degree.

Table 1. Reasons to Major in Computing

| Reason | BS in Computing | BS in other major | left w/no degree |
|---|---|---|---|
| Attracted by the challenge of a difficult curriculum | 50% | 25% | 0% |
| Good at math or science | 86% | 75% | 25% |
| High School advisor or teacher recommended it | 14% | 0% | 0% |
| Like to solve problems | 64% | 75% | 0% |
| Friends recommended it | 4% | 0% | 0% |
| Participated in camp or workshop that influenced me | 21% | 0% | 0% |
| Parent, other relative, or friend is an engineer | 54% | 63% | 50% |
| Parents, siblings, or other relatives recommended it | 39% | 63% | 25% |
| Anticipated possibility of good college scholarship | 14% | 0% | 0% |
| Wanted to be able to get a good paying job after I graduate | 68% | 63% | 0% |
| Wanted to use computing to address social problems | 7% | 13% | 0% |

One of the students who did not complete a degree indicated that she was "forced by parents" to major in computing, another mentioned the movie "Hackers" as inspiration to study computing. The free form answers provided by those who completed the degree had more to do with their prior experience with programming, summer programs, or the content of the program offered at SCU.

### 4.2 Preparation and Initial Confidence

We asked the students whether they felt their high school had prepared them for success in their computing major. Half of those who did not complete a degree said "no," and 75% of those who graduated in another major stated their high school had not properly prepared them. Just over half (54%) of those who completed a degree in computing felt their high school DID prepare them for success in the major.

In response to a question about their self-efficacy in computing at the beginning of their first year in college, 75% of those who did not complete a degree felt "fairly" or "very" confident of success in getting their degree in computing. Also, 75% of those who completed their major in another discipline had felt fairly or very confident about success in the computing degree program. Both of these groups were more skewed toward "fairly" than "very," while the group who completed the degree in computing were skewed a bit toward "very confident" in the 82% who were fairly or very confident they would succeed.

### 4.3 Participation in the Engineering Community

We provided a list of activities and asked about the women's participation in their most recent year of study. Table 2 presents the activities in question, and the percentage of students who responded that they were involved at least once in their last year of enrollment.

Table 2. Participation in Engineering Activities

| Activity | BS in Computing | BS in other major |
|---|---|---|
| Professional Eng. society | 53.6% | 37.5% |
| Minority Eng. society | 71.4% | 25% |
| Minority or Multi-cultural activities | 38.5% | 12.5% |
| Women in Eng. events | 66.7% | 37.5% |
| Departmental events | 74.1% | 50% |
| Design competitions | 26.9% | 25% |
| Undergraduate research | 12.5% | 37.5% |
| Co-op or internship | 50% | 25% |

The students who left the university without a degree either answered that they did not participate in ANY activity, or they did not respond to the question at all (leading one to assume they did not participate?). Half of the students who completed a degree in a different major did not participate in any of the activities mentioned, while only 7% of those who completed a degree in computing were not involved in any activity. One woman commented that she was unaware of these opportunities while a student.

### 4.4 Perceived Impact on Retention

Several issues were identified to be of possible influence in student retention. We asked the women to indicate on a scale of 0 to 4, the level of influence that each of these factors had on their decision to stay with or leave the major. Unfortunately, none of the students who left the university without a degree answered these questions. Table 3 presents the average impact level of each factor for the students who completed a degree in computing and those who completed a degree in another major.

Table 3. Perceived influence (scale 0-4) on retention decision

| Factor | BS in Computing | BS in other major |
|---|---|---|
| Availability of financial aid or scholarship | 2.04 | 0.00 |
| Department faculty showed interest in me | 2.96 | 1.50 |
| Reasonable workload | 2.07 | 0.88 |
| Friendly climate in classes | 2.46 | 1.13 |
| Grades earned | 2.75 | 1.88 |
| Faculty helped me understand the profession | 2.25 | 1.50 |
| Quality of teaching | 2.86 | 1.00 |
| Effectiveness of advising | 2.61 | 1.75 |
| Availability of jobs and/or internships | 2.46 | 1.00 |
| A good personal “fit” for me | 2.79 | 1.88 |
| Confident of success in classes | 2.67 | 1.88 |
| Positive interactions with other students | 3.04 | 1.25 |
| Positive design team or collaborative experiences | 2.93 | 1.00 |

It is interesting that many students felt none of these factors had any influence at all on their decision to change majors, and a couple of students also claimed no influence on their decision to remain in the major.

Some students left because their interests lay elsewhere - the best reason of all to leave a program. However, some did indicate a less than welcoming attitude from a few individuals.

The women were asked in an open-ended question to provide the one biggest factor that helps/helped you persist in your study of computing. From analysis of the responses, several themes emerged relating to intrinsic motivations, social supports, academic supports, and potential professional opportunities. There were forty-five respondents to this question, some who answered with single words and some who wrote extended responses. What follows are summaries and quotes, from the forty-five women who responded, grouped within four general themes that emerged with overlap occurring in some comments.

#### 4.4.1 Intrinsic Motivators

These themes emerged from comments that indicated some form of internal drive such as an interest in computing because of the nature of the field, the kind of work involved, the challenge provided, the excitement they feel when engaged in computing, and the personal rewards they experience and their development.

The comments included those who exuberantly expressed passion for the field and a determination to complete the program for personal achievement. With comments such as:

"I feel like the effort I put into it is personally rewarding."

"I love it."

"It is appealing and interesting."

"I wanted to prove to myself that I could do it."

"It never occurred to me to quit."

#### 4.4.2 Social Supports

Several of the women indicated social support included family, friends, classmates, and personal interaction with faculty. These forms of support indicated caring, friendships among classmates, faculty extending themselves to demonstrate personal considerations to the students' success.

Some of the comments in this regard included:

"I got to know all of my professors personally, they really help me understand the material as schedule office hours around my busy schedule as a student athlete. I have found a group of close friends in all of my classes that are just as motivated as me as I am to study and get work done which also helps to get through understanding every difficult concept."

"Without my friends, I would not have graduated."

#### 4.4.3 Academic Supports

The comments that were indications of academic support included financial aid, tutoring, out-of-class time with instructors, advisors, study groups, and design of curriculum.

Some of the comments in this regard included:

"I loved the other classes associated with the web design major and my computing abilities were good enough to survive the COEN classes."

"Exciting, hands-on coursework/lab work"

"…feeling like the professors were truly invested in helping their students succeed and pro-actively participating in their development."

#### 4.4.4 Potential Professional Opportunities

The comments that were indications of professional opportunities were those that included the potential for quality employment, interesting work, and the ability to contribute to innovations through computing.

Some of the comments in this regard included:

"Tech is integral to our future and is a major component in education. I stay on top of tech and continue to have a strong interest in it because the more knowledgeable I am, the better tutor/counselor/mentor I can be for my students."

"Good job after education, relevant senior project that would be applicable to real life"

"Internships and knowing all the possibilities of different jobs after I complete my degree."

"Knowing that this is something I could do for the rest of my life…I decided to play to my strengths and switch to a design-oriented major."

"To be honest, money and challenge. I am practical. I wanted a career where I could afford what I want, and I wanted one that would challenge me day-to-day."

"Having a chance to apply the skills I gained helped me to see how widely applicable my skills were. This sparked my interest—I love having something to apply my learning to."

There were some comments that reflected more than one theme. For example, one woman described how she enjoyed the web programming group projects where they were able to select partners, groups, and projects, as well as the engineering project. She found that she got to learn to interact with people, how to better understand her strengths, and which phases of technology creation she enjoyed.

There was one woman who indicated that she persisted in her study of computing because she had begun the curriculum and would not have been able to complete another degree within four years if she switched.

## 5 Summary of Lessons Learned

Surveys are not the best way to gauge impact of programs, though they do help gauge the perception of that impact. We had done previous studies showing the difference in completion rates of those participating in an intervention vs. those who didn't [2, 3]. We found that those who participated in the intervention completed their engineering degrees at substantially higher rates than those who did not participate. However, it is difficult to decipher what it was about the intervention that made the difference. Hoping to get a better handle on this, we asked the women what they felt had the most impact on their decisions.

One of the problems with the survey was that (although we asked them to rate the impact of only those activities in which they participated) many women replied that an activity "had no impact" when, in fact, they had not participated in it. As a result, we do not try to draw conclusions from this part of the survey. Of most value were the free form comments from students, which overwhelmingly reinforced the notion that "connection" is extremely important.

We plan to follow up with an analysis of the results from students who had not yet graduated when they took the survey. It will be interesting to see if any of their answers on the survey may be predictive of their eventual graduation in the computing major or another major on campus. We will have to wait until at least 2016 (for four year graduation rates) and a until 2018 if we need to follow these students longer for graduation results.

## 6 References

[1] DAVIS, Ruth E. "Community-based Learning Develops Diverse Engineering Workforce." Innovations 2005 – Special Edition: World Innovations in Engineering Education and Research , 1(2005), chapter 2.

[2] SULLIVAN, Kieran, with Ruth Davis, "Increasing Retention of Women Engineering Students," Proceedings of the 2007 ASEE Annual Conference and Exposition, Honolulu, HI, June, 2007.

[3] DAVIS, Ruth, with Kieran Sullivan, "Why Do/Don't They Stay: Preliminary Results of an Engineering Retention Study," Proceedings of the International Conference on Engineering Education and Research, Melbourne, Australia, December, 2007.

[4] Krishnan, Shoba and Ruth E. Davis, "Engineering Projects for the Community," in Proceedings of the 2011 International Conference on Engineering Education, International Network for Engineering Education and Research, August 20-26, 2011, Belfast, Northern Ireland.

[5] Krishnan, Shoba, with Ruth Davis, Marco Bravo, and Melissa Gilbert, "Projects Integrating Sustainability and STEM Outreach," ASEE PSW Region Conference, March 31 – April 2, 2011, Fresno, CA.

[6] Davis, R., Krishnan, S., Nilsson, T., & Fylling Rimland, P., "IDEAS: Interdisciplinary Design Engineering and Service", International Journal for Service Learning in Engineering, Humanitarian Engineering and Social Entrepreneurship, Special Issue: University Engineering Programs That Impact Communities: Critical Analyses and Reflection, January 2015, pp. 165-179.

# Identifying and Addressing the Challenges of Industry-Academic Partnerships at Minority-Serving Institutions

**A. Qi'Anne Knox**[1], **B. A. Nicki Washington**[1]
[1]Computer Science, Howard University, Washington, DC, USA

**Abstract** - *Computer science has suffered declining student degree attainment for the past decade. At Howard University, a novel program was piloted with Google, known as the Googler-in-Residence (GIR) program. Through the program, which is currently in its second year, a number of challenges were identified that will potentially face other industry-academic partnerships, particularly at minority-serving institutions (MSIs).*

*The goal of this paper is to identify those challenges, through the results of the GIR program, and provide guidance for universities and industry affiliates on how to best partner to prepare students of color in undergraduate computer science departments, particularly at MSIs. As a result of this work, a clear blueprint on how to best develop industry-academic partnerships at MSIs is provided that ensures both industry affiliates and universities benefit fully from the strategic partnership.*

**Keywords:** Minority-Serving Institution, Googler-In-Residence, hackathons

## 1 Introduction

Computer science has suffered declining student participation for the past decade [3]. In recent years, this has been a topic of focus for both universities and industry affiliates. With the minority population growing through the year 2020, it is imperative that more students of color are introduced to and complete the CS pipeline, leading to careers in CS [6][11][12][14-16]. A variety of initiatives are forming to counter this decline and introduce more students to computer science programs at the secondary and post-secondary levels [4][5][7][8].

At Howard University, a novel program was piloted with Google, known as the Googler-in-Residence (GIR) program. The pilot program partnered Google with the Department of Computer Science to provide Google software engineers as full-time faculty members within the department for one semester. GIRs taught Introduction to Computer Science courses, in an attempt to address some of the problems faced by students of color in introductory computer science courses within the major.

This effort was part of a multi-phased program, designed to restructure the first two years of study for undergraduate students, to better prepare them for the major and careers in computer science. Through the program, which is currently in its second year, a number of challenges were identified that will potentially face other industry-academic partnerships, particularly at minority-serving institutions (MSIs). Given the unique opportunity to directly impact the CS pipeline, MSIs should be a natural choice for industry affiliates to focus efforts on diversifying the CS pipeline [9]

The goal of this paper is to identify those challenges, through the results of the GIR program, and provide guidance for universities and industry affiliates on how to best partner to prepare students of color in undergraduate computer science departments, particularly at MSIs [13]. As a result of this work, a clear blueprint on how to best develop industry-academic partnerships at MSIs is provided that ensures both industry affiliates and universities benefit fully from the strategic partnership.

The remainder of this paper is organized as follows. In section 2, the motivation behind the GIR program is discussed. In section 3, results from the GIR program are presented. In section 4, the significant challenges identified as a result of the program are discussed. Section 5 summarizes the work and provides future opportunities.

## 2 Motivation

The Googler-In-Residence (GIR) program was one component of a multi-phase program, designed to increase the retention of African-American CS undergraduates in the first two years of study [17].

To accomplish this goal the following objectives were outlined:

1. Improve student performance in the first three CS courses in CS program study
2. Improve students' perceptions of CS
3. Improve department-wide support mechanisms for all classifications

### 2.1 Improve Student Performance in the first three CS courses in CS Program Study

The department has restructured the four-year program of study to require freshmen to complete a new course, CS0-

Intro to SCS (fall-freshmen year), followed by CS1 (spring-freshmen year), and CS2 (fall-sophomore year). The new implementation of CS0 was designed to provide students with a “big picture” of CS, the applications and career paths available, and introduce CS fundamentals such as algorithms, problem-solving, and computational thinking prior to learning programming. This course would also provide an introduction to programming via Python, in preparation for CS 1, which is taught using C++.

## 2.2 Improve Students’ Perceptions of CS

The GIR program was designed specifically to help address this goal. By leveraging the Black Googlers Network, Googlers were full-time faculty members in the department for one or two semesters. This allowed them to not only teach classes, but also work with students outside of class for office hours and other department programs.

## 2.3 Improve Department-wide Support Mechanisms for all Classifications

Interventions such as peer-tutoring/mentoring sessions were provided throughout the academic year. In addition, the department, in partnership with Google, hosted several events, including hackathons, mock interviews, game nights, and other events to help expose students to CS and encourage increased interaction within the department, including faculty, staff, and students.

The GIR program was one of several components of the program, which consisted of the following:

1. Pre-Freshmen in Computer Science (PREFICS) Virtual Pre-College Program (pre-college component)
2. Introduction to Computer Science Course curriculum restructure
3. HU-Google Mentor Program
4. CS Bootcamp
5. Undergraduate Researchers

# 3 Results of GIR Program

Table 1 presents the course enrollment for both courses in the 2012-2013 and 2013-2014 academic years. It should be noted that SCS freshmen enrolled in the CS 0 course in the fall semesters and CS 1 in the subsequent spring semester. It should also be noted that CS 0 was only offered in the fall semester of the 2012-2013 academic year. CS 0 experienced an 83% enrollment increase in the fall 2013 semester. In addition, the success of the course resulted in it being offered in the spring 2014 semester. In total, a 297% enrollment increase was experienced. CS 0 was the course taught by the GIRs.

**Table 1. Course Enrollment –CS 0 and CS 1**

| | Fall 2012 | Spring 2013 | Fall 2013 | Spring 2014 | Fall 2014 | Spring 2015 |
|---|---|---|---|---|---|---|
| **CS 0** | 25* | N/A | 55* | 64 | 60** | - |
| **CS 1** | 29 | 33* | 35 | 48* | - | 47* |
| **CS 2** | 18 | 29 | 21** | - | 34** | - |

**Table 1. Course Enrollment-CS 0, 1, and 2**
**** indicates correct sequence for SCS majors***

Table 1 presents the course completion rates for CS 0, CS 1, CS 2 from the 2012-2013 academic year to date. Approximately 94% of all SCS students passed CS 0 in the fall 2013 semester, compared to 72% in the fall 2012 semester. For CS 1, approximately 67% of students passed the course in the spring 2014 semester, compared to 59% in the spring 2013 semester. These results indicate that a higher number of students passed the first two CS courses of study in the major than in the year prior to implementation of the new program.

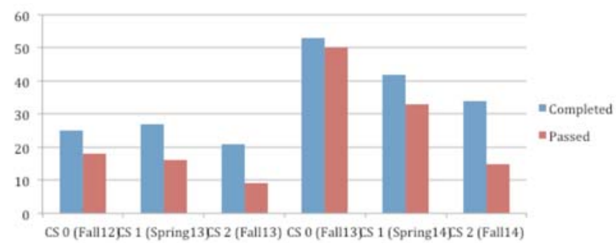



**Figure 1. Course Completion and Passing-CS 0,1, and 2.**

Figure 1 illustrates the course completion and passing rates for CS 0, 1, and 2. While students are performing better in CS 0 and 1, there are still issues with students passing CS 2 that haven’t been addressed. This signals challenges that haven’t been addressed.

In the 2014-2015 academic year, the GIR program expanded to four additional HBCUs: Hampton University, Fisk University, Spelman College, and Morehouse College. Each university had at least one GIR for one semester serving as a full-time faculty member.

# 4 Challenges

While the GIR program has been a success in attracting students to CS, engaging them, and teaching them, there were still challenges identified that clearly illustrate the need for strategic partnerships between industry and academia, specifically in the context of minority-serving institutions such as HBCUs, Hispanic-Serving Institutions (HSIs), and Tribal Colleges and Universities (TCUs). These challenges are critical to not only identify, but also begin to address as universities form partnerships with industries.

### 4.1 Programming vs. Problem-Solving

A major challenge was finding the correct balance between content that focused on programming and problem solving in CS 0. As industry partners, it was natural for Googlers to heavily focus on programming. Because students receiving internships with the company must be able to program in some language by the time they interview for summer internships (second semester-freshman year), it was imperative that students learn programming in their first semester. Python was chosen as a high-level language to introduce students to programming in CS 0. However, the introduction of this material came at the expense of less focus on problem solving.

Given that a large majority of students within the department had no prior experience in CS, there was a strong push from faculty to focus on problem solving. This was a skill that was noted as lacking in subsequent courses such as CS 2. This was also what was originally attributed to the lower rate of course enrollment and completion in CS 2, as illustrated by Table 1 and Figure 1. Given the low percentage of students of color exposed to computer science, or enrolled in higher-level math courses in high school, most students enter universities with little to no knowledge about computer science, including core computing fundamentals such as computational thinking, algorithm development, problem solving, and more. It is imperative that a proper balance of problem solving be incorporated, prior to and during the teaching of programming concepts.

### 4.2 Instructor Professional Development

This was the first time that most of the GIRs served as instructors in a formal setting. As a result, there were a number of unexpected activities and challenges encountered. It was initially assumed that each GIR would have a standard curriculum to teach the course, to maintain consistency across each university. Due to logistical challenges, this was not in place prior to the start of the 2014-2015 academic year. In addition, each university is a unique environment, which requires navigating administration, colleges, and departments.

In addition to being their first time serving as instructors, many of the GIRs did not attend HBCUs. This was their first time in such an environment. Understanding the needs of students of color, specifically in the context of computer science, is something that should've been provided to participants via the department. This included information on the department, student demographics, and also the importance of a culturally-relevant curriculum. As students of color traditionally do not choose computer science as a major, due to assumptions that it is more for White and Asian males, it is imperative that all instructors are aware of the importance of making CS content relevant to the daily lives of all students[1][2].

### 4.3 Proper Credit for Residents

The commitment to teach full-time for one semester is a significant sacrifice for GIRs. This means that individuals are removed from industry for that time, to focus on their teaching load, which can range from one to two courses. Part of the challenge in finding GIRs was the concern regarding receiving proper credit for participating in the program. Residents must be properly incentivized to participate, and not view the program as a hindrance to their current workload or career trajectory. Instead, there should be clear incentives (e.g. certifications, recognition, positions, etc.) for participants to see the value in the experience. While this must come from mainly from the industry affiliate, universities can provide additional incentives by providing adjunct professor positions for residents.

### 4.4 Sustainbility

One of the major challenges with industry-academic partnerships is sustainability. The GIR program is currently in its second year of implementation, across five HBCUs. Expanding to additional schools requires more Googlers willing to take a full semester of leave to participate, receive professional development, and serve as a full-time instructor. With the number of HBCUs alone, the ability to provide a GIR at each institution may be infeasible.

Proper measures must be in place to provide a program that will be of benefit to the students, faculty, and Googlers, but also ensure that it is sustainable for years to come. Currently, the research team is working with Google to identify the best methods for sustaining the program, including the use of virtual GIRs and team-teaching courses with department faculty. This would allow a single GIR to reach more students across more institutions.

In addition, CS education students at both the graduate and undergraduate levels should be leveraged, when possible, to ensure course continuity in subsequent semesters when the GIR is not present. This can be supported through practicum courses and stipends for graduate students.

## 5 Conclusions

Industry-academic partnerships are not new. However, with the push to diversify computer science, these partnerships are looking for unique ways to reach students who traditionally have been underrepresented. The GIR program is a novel program designed to bridge industry and academia, by providing industry personnel teaching experience and undergraduate students direct access to industry representatives that provide a real-world context for the computing fundamentals they learn in the classroom. These partnerships must be well-planned and executed at minority-serving institutions, to ensure that both parties benefit from it. Proper planning includes knowing potential challenges, such as those previously mentioned, and addressing them prior to implementation, to ensure that not only are industry personnel are adequately prepared to enter the environment, but students are also best prepared to succeed both in and out of the classroom.

# 6 References

[1] Fleming, L. et al. 2006. "Why Students Leave Engineering: The Unexpected Bond", Proceeding of the 2006 American Society of Engineering Education Conference and Exposition.

[2] Furr, S. and Elling, T. 2002. "African-American Students in a Predominantly-White University: Factors Associated with Retention." College Student Journal, vol. 36 no. 2.

[3] Gibbons, M. 2009. "Engineering by the Numbers". American Society for Engineering Education.

[4] Goode, J. 2010. "Connecting K-16 Curriculum & Policy: Making Computer Science Engaging, Accessible, & Hospitable for Underrepresented Students." Proceedings of the 40th SIGCSE Technical Symposium on Computer Science Education.

[5] Goode, J. 2008. "Computer Science Segregation: Missed Opportunities." *The Voice. 4*(2).

[6] Jackson, S.A. "The Quiet Crisis: Falling Short in Producing American Scientific and Technical Talent." Building Engineering and Science Talent.

[7] Margolis, J., Estrella, R., et.al. Stuck in the Shallow End: Education, Race, and Computing.

[8] Margolis, J., Goode, J., Bernier, D. 2011. "The Need for Computer Science." Education Leadership, vol. 68, no.5.

[9] May, G. and Chubin, D. 2003. "A Retrospective on Undergraduate Engineering Success for Underrepresented Minority Students. Journal of Engineering Education.

[10] Members of the 2005 Committee on Science, Engineering, and Public Policy. 2010. Rising Above the Gathering Storm Revisited: Rapidly Approaching Category.

[11] National Academy of Engineers. The Engineer of 2020 Project Prospectus.

[12] National Science Foundation, Division of Science Resources Statistics. 2011. Women, Minorities, and Persons with Disabilities in Science and Engineering.

[13] Taulbee Survey. Computing Degrees and Enrollment Trends.

[14] The President's Council of Advisors on Science and Technology. 2010. Prepare and Inspire: K-12 Education in Science, Technology, Engineering, and Math (STEM) for America's Future.

[15] Toossi, M. 2006. "A New Look at Long-term Labor Force Projections to 2050." Monthly Labor Review.

[16] US Census Bureau. 2004. "US Interim Projections by Age, Sex, Race, and Hispanic Origin."

[17] Washington, A. N., Burge, L., Mejias, M., Jean-Pierre, K., and Knox, Q. "Improving Undergraduate Student Performance in Computer Science at Historically Black Colleges and Universities (HBCUs) through Industry Partnerships," *Proceedings of the ACM Special Interest Group on Computer Science Education (SIGSCE)*, 2015, USA.

# Framework to Identify At-Risk Students in E-learning Environment

Alpa Reshamwala, and Sunita Mahajan

***Abstract*— Fuzzy logic based methods together with the techniques from Artificial Intelligence have gained importance. It provides a sound foundation to handle imprecision and vagueness as well as mature inference mechanisms using varying degrees of truth. Sequential pattern mining is the mining of frequently occurring ordered events or to discover frequent subsequences as patterns in a sequence Database. Frequent sequential pattern discovery can essentially be thought of as association rule discovery over a temporal database. The time interval sequential patterns provide more valuable information than a conventional sequential pattern. However, this approach may cause a sharp boundary problem. In this paper, we aim to introduce a new algorithm "FTI-Apriori-Event" which predicts an optimum fuzzy time interval for the future occurrence of a given event based on fuzzy set as they provide a smooth transition between member and nonmember of a set, which in turn helps in "providing right products at the right time to the right customers".**

## I. Introduction

SEQUENTIAL Pattern Mining finds remarkable sequential patterns among the huge database. It finds out recurrent subsequences as patterns from a sequence database. With huge amounts of data constantly being gathered and warehoused, many industries are becoming involved in mining sequential patterns from their database. Sequential pattern mining is one of the most well known technique and has extensive applications Many approaches have been proposed to mine information and mining sequential patterns is one of the most significant ones [1] [2] [3]. For example, having bought a printer, a customer comes back to buy a printing papers and an ink cartridge refill kit next time. With the help of this pattern, the retailer can send an ink cartridge refill kit campaign program to customers after customers bought an inkjet printer, an ink cartridge, and printing papers. The retailer can use such information for analyzing the behavior of the customers, to understand their interests, to satisfy their demands, and above all, to predict customers needs. In the medical field, sequential patterns of symptoms and diseases exhibited by patients identify strong disease correlations that can be a precious source of information for medical diagnosis and defensive medicine. The exploring behavior of a user can be extracted, in Web log analysis from member records or log files. For example, having viewed a web page on "Data Mining", user will return to explore "Business Intelligence" for novel information next time.

Alpa Reshamwala is Assistant Professor in Computer Engineering Department, MPSTME, SVKM's NMIMS University, Mumbai, India. (alpa.reshamwala@nmims.edu).

Dr. Sunita Mahajan, is Principal in Institute of Computer Science, M.E.T, Bandra, Mumbai, India (e-mail: sunitam_ics@met.edu).

These sequential patterns yield huge benefits, when acted upon, increasing customer royalty. Hence it's very important for an efficient sequence-mining algorithm to predict future customer behavior. The problem of mining sequential patterns was first presented by Agarwal and Srikant [1], which discover patterns that occur recurrently in a sequence database. Later in mid 1990's, following Agrawal and Srikant [1], several scholars offered more efficient algorithms [4] [5] [6] [7]. Besides these, works have been done to extend the mining of sequential patterns to other time-related patterns. Existing approaches to find appropriate sequential patterns in time related data are mainly classified into two approaches. In the first approach developed by Agarwal and Srikant [1], the algorithm extends the well-known Apriori algorithm. This type of algorithms is based on the characteristic of Apriori that any subpattern of a frequent pattern is also frequent [1]. The later, uses a pattern growth approach [4], employs the same idea used by the Prefix-Span algorithm.

Since the early eighties, due to rapid industrialization and economic growth, engineering and technical education in India have been developing faster than anywhere else in the world, and India now has the second largest number of engineering students in the world [8]. One of the biggest challenges that higher education faces is to improve student retention [9, 10]. Student retention has become an indication of academic performance and enrolment management. Increasing student retention is a long-term goal in all-academic institutions. To retain students, it is necessary for the higher education (HE) institutions to have the ability to predict students who are 'at-risk' in there programmed of study as early as possible. Timely intervention, based on early identification of 'at-risk' students is likely to help improve their performance so that they can progress and complete their study successfully.

Since four decades, a sincere concern has aroused among managerial, professional, towards the satisfaction of teaching-learning objective in Academia. In educational data mining the decision-making bodies of the institutions have been analyzing exhaustive data sets of students in successive batches for implementing various machines learning task. Research in education data mining, till date, shows that the decision-making dimensions revolve merely around over all students appraisal in their course of participation. In this context, many feature selection models have been developed since two and half decades. It has been found that there feature selection model varied according to the criterion function used to satisfy the machine-learning task.

Sequential pattern mining [1], used to discover frequent sequential patterns which are lists of Web pages ordered by viewing time for predicting visit patterns. In EDM, "Take Action before it's too late", Imagine being able to change a student's path if you were aware of risk factors early on. Faculty will be able to help students when they first start showing signs of falling behind or disengaging in their courses. With students browsing patterns or sequence of web click patterns make it easy for faculty to see immediately who's not participating, missing deadlines, and/or performing below average. By supervising student's performance and involvement in a course, as well as examining how this relates to grades, faculty can potentially spot areas of the course to improve. Such improvements in the course allow for the continual improvements that accrediting bodies are recommending.

In this paper, the concept of fuzzy set theory is used so that fuzzy time-interval student's event sequences are discovered on Learning Management System. The dataset taken for research is the raw data of Blackboard LMS of undergraduate Students of SVKM's NMIMS, Mukesh Patel School of Technology Management and Engineering. The dataset is about 14 lacs with 2725 students, 1140 different web clicks events on LMS enrolled in a course at the university for the period of 3 months i.e. from January 2014 to March 2014. We aim to introduce a new model by implementing algorithm "FTI-Apriori-Event" which predicts an optimum fuzzy time interval for the future occurrence of a given event based on fuzzy set as they provide a smooth transition between member and nonmember of a set, which in turn helps in "providing right products at the right time to the right customers". In LMS the algorithm results in predicting the optimal fuzzy time at which students will perform a given task. The complement list of the above results will predict the students list which are at risk and do not perform the given task in the optimal time detected.

## II. Related Work

E-Learning is relatively new emerging technology that spans the universities and other institutions. It concentrates on utilizing Knowledge of teachers in a way that academic courses can be delivered anywhere and anytime. As time passes, the growth of knowledge imposed new requirements for a better use of this asset. Knowledge Management (KM) is the way to a better utilization of the knowledge. The emergence of Information Technology (IT) provides tools for such management, until the new era of learning, namely the E-Learning, poses its impacts on knowledge acquisition and utilization, and hence on KM. This new trend in both KM and E-Learning lets many educational institutional organizations in most parts of the world take the advantages of these technologies. With students browsing patterns or sequence of web click patterns make it easy for faculty to see immediately who's not participating, missing deadlines, and/or performing below average. By supervising student's performance and involvement in a course, as well as examining how this relates to grades, faculty can potentially spot areas of the course to improve. Some researchers developed a framework that identifies students with higher risk of dropping out at an early stage to allow institutions to provide assistance for students in need; thus increasing the quality and quantity of students in higher education. Sequential pattern mining discovers frequent lists of Web pages ordered by viewing time for predicting visit patterns of the students in an e-learning environment.

Mingming Zhou in [11], presents the application of sequential data mining algorithms to analyze computer logs to profile learners in terms of their learning tactic use and motivation in a web-based learning environment (gStudy). P. A Patil and R. V. Mane, in [12], proposes Generalized Sequential Pattern mining algorithm for finding frequent patterns from student's database and Frequent Pattern tree algorithm to build the tree based on frequent patterns. This tree can be used for predicting the student's performance as pass or fail. Thomas, E.W *et. al.* in [13], discusses an discriminant analysis to predict student performance in an introductory electro magnetism course at Georgia Tech and also discuss how discriminant analysis can be used as an evaluation technique. We show that this approach provides results that are both more interpretable and statistically sound than traditional measures. Patrick D. Schalk *et. al.*, in [14], developed instruments designed to measure aptitude in mathematics to develop a machine learning-based predictive model for student performance. Mamta Singh *et. al.* in [15], proposes a framework for identify the most contributed attributes towards academia, for the performance of second year students of computer science and application course. An appropriate supervised machine-learning model is applied upon our set of inherent attributes in order to arrive (NBC) at predictive scenarios for given pattern of external attributes. B. M. Bidgoli *et. al.* in [16], presents an approach to classify students in order to predict their final grade based on features extracted from logged do to in an education web-based system using GA. K V K Kishore *et. al.* in [17], used Multilayer Perceptron (MLP) based prediction application and Alkhasawneh, R. and Hobson, R. in [18], we developed two neural network models using a feed-forward backpropagation network to predict retention for students in science and engineering fields. Rizvi, M *et. al.*, in [19], to determine the effectiveness of a CS0 course using Scratch for improving the retention, the performance and the attitudes of at-risk majors is discussed. Farhana Sarker *et. al.*, in [20], predicts performance of logistic regression approach and also suggest that logistic regression performs better for identifying 'at- risk' students in their programme of study. Xiaojing Wang *et. al.* in [21], proposed and defined a degree to which the given estimate is accurate and applied in predicting at-risk students. Hai-yan Wu *et. al.* [22], present a method of transaction identification based on users' access path tree. It can find out all the transactions effectively and can also make some improvements on PrefixSpan algorithm.

Zadeh first proposed the theory of fuzzy sets in 1965[23]. Many researchers have exploited Fuzzy technique to mine sequential patterns or association rules from databases. In data mining fuzzy concepts are applied in clustering, association rules, function dependencies, data summarization, web applications and image retrieval [24]. Fuzzy is represented as: item's quantity [25], quantitative attribute [26][27] as product taxonomies or in generalization [28], importance of items [29], as transactions [30], support and confidence measure [31], domain partitions [32] and a technique to represent rule's interestingness [33-36]. Some have represented as time intervals as distribution of train's arrival time [37], Chang et al. [38], proposed an algorithm called sequential pattern mining with fuzzy time intervals (SPFTI) and the experimental result verified SPFTI algorithm outperforms sequential patterns mining with fixed time interval. In [39], Chang et al. proposed an algorithm called integrated sequential pattern mining with fuzzy time intervals (ISPFTI) with apriori property.

In this paper, we aim “to predict for a given event,

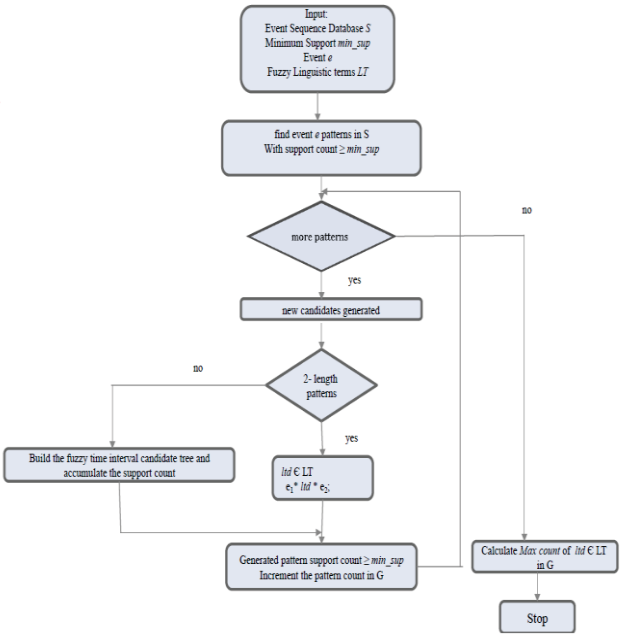



Fig. 1. FTI-Apriori-Event Algorithm

which all students are regularly submitting on time and which are submitting late”. In LMS the “FTI-Apriori-Event” algorithm, results in predicting the optimal fuzzy time at which students will perform a given task. The complement list of the above results will predict the students list which are at risk and do not perform the given task in the optimal time detected.

## III. FTI-Apriori-Event Algorithm

In this section, we aim to propose a novel algorithm “FTI-Apriori-Event” which predicts an optimum fuzzy time interval for the future occurrence of a given event. On execution of FTI-Apriori-Event algorithm, the algorithm not only predicts an optimum time for a particular event but also generates sequence list where these optimum fuzzy time interval sequence can occur in future.

On applying the above algorithm on the dataset of Table 6.9, as shown in Figure 6.13, we get L1 as input event “a” with minimum support greater than or equal to minimum support of 10%. Then the following candidate time-interval sequences exist in C2: (a LT1 a); (a LT2 a) and (a LT3 a) with support of 30%, 30% and 20% respectively. In summary, C2 can be generated as L1 X LT X L1; where X denotes join. L2 will be the C2 satisfying the minimum support threshold. Next, similarly generate Ck and Lk. Later, we calculate the frequent fuzzy time interval in the Lk by implementing MaxCount(); which returns the optimal fuzzy time interval for those discovered sequences.

## IV. Results and Discussions

For the experimental evaluation and performance study of the FTI-Apriori-Event algorithm, we have implemented FTI-Apriori-Event in Sun Java language and tested on an Intel Core Duo Processor, 2.10 GHz with 4GB main memory under Windows 7 operating system.

Consider dataset of the raw data of LMS Blackboard for three months i.e 90 days. Hence, by taking time interval of 90 days, we get values of 30, 15 and 7 days partitions. The detailed fuzzy time interval partitions and the fuzzy membership functions are as shown below,

a. For time interval of 30 days in three months’ time (90 days), we get three fuzzy time interval partitioned sets as, FTI- 3 {LT0; LT1; LT2} where; the literals can be represented by three linguistic terms: Instant (I), Immediate (Im) and Later (L). The fuzzy membership function can be as given by equations (1), (2), and (3) respectively.

$$\mu_{Instant}(t_{ij}) = \begin{cases} 1 & ; tij = 0 \\ \frac{30-tij}{30} & ; 0 < tij \leq 30 \\ 0 & ; tij \geq 30 \end{cases} \quad (1)$$

$$\mu_{Immediate}(t_{ij}) = \begin{cases} \frac{tij}{30} & ; 0 < tij < 30 \\ \frac{60-tij}{30} & ; 30 < tij \leq 60 \\ 0 & ; tij \geq 60 \\ 1 & ; tij = 30 \end{cases} \quad (2)$$

$$\mu_{Later}(t_{ij}) = \begin{cases} \frac{tij-30}{30} & ; 30 < tij < 60 \\ \frac{90-tij}{30} & ; 60 < tij \leq 90 \\ 0 & ; tij \leq 30 \text{ or } tij \geq 90 \\ 1 & ; tij = 60 \end{cases} \quad (3)$$

The graphical representation of the above fuzzy membership functions are shown in Figure 2.

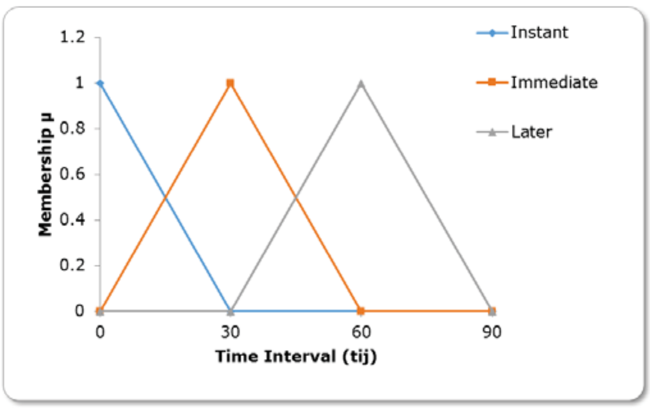



Fig. 2. Membership functions of Fuzzy time interval FTI-3

b. For time interval of 15 days in three months’ time (90 days), we get six fuzzy time interval partitioned sets as, FTI- 6 {LT0; LT1; LT2; LT3; LT4; LT5} where, literals can be represented by linguistic terms as shown in (a). The fuzzy membership function can be as given by equations (4), (5), (6), (7), (8) and (9) respectively.

$$\mu_{LT0}(t_{ij}) = \begin{cases} 1 & ; tij = 0 \\ \frac{15-tij}{15} & ; 0 < tij \leq 15 \\ 0 & ; tij \geq 15 \end{cases} \quad (4)$$

$$\mu_{LT1}(t_{ij}) = \begin{cases} \frac{tij}{15} & ; 0 < tij < 15 \\ \frac{30-tij}{15} & ; 15 < tij \leq 30 \\ 0 & ; tij \geq 30 \\ 1 & ; tij = 15 \end{cases} \quad (5)$$

$$\mu_{LT2}(t_{ij}) = \begin{cases} \frac{tij-15}{15} & ; 15 < tij < 30 \\ \frac{45-tij}{15} & ; 30 < tij \leq 45 \\ 0 & ; tij \leq 15 \text{ or } tij \geq 45 \\ 1 & ; tij = 30 \end{cases} \quad (6)$$

$$\mu_{LT3}(t_{ij}) = \begin{cases} \frac{tij-30}{15} & ; 30 < tij < 45 \\ \frac{60-tij}{15} & ; 45 < tij \leq 60 \\ 0 & ; tij \leq 30 \text{ or } tij \geq 60 \\ 1 & ; tij = 45 \end{cases} \quad (7)$$

$$\mu_{LT4}(t_{ij}) = \begin{cases} \frac{tij-45}{15} & ; 45 < tij < 60 \\ \frac{75-tij}{15} & ; 60 < tij \leq 75 \\ 0 & ; tij \leq 45 \text{ or } tij \geq 75 \\ 1 & ; tij = 60 \end{cases} \quad (8)$$

$$\mu_{LT5}(t_{ij}) = \begin{cases} \frac{tij-60}{15} & ; 60 < tij < 75 \\ \frac{90-tij}{15} & ; 75 < tij \leq 90 \\ 0 & ; tij \leq 60 \text{ or } tij \geq 90 \\ 1 & ; tij = 75 \end{cases} \quad (9)$$

The graphical representation of the above fuzzy membership functions are shown in Figure 3.

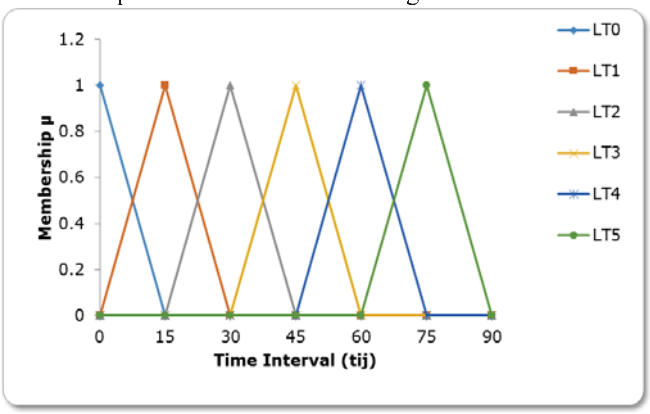



Fig. 3. Membership functions of Fuzzy time interval FTI-6

c. For time interval of 7 days in three months' time (90 days), we get thirteen fuzzy time interval partitioned sets as, FTI- 13 {LT0; LT1; LT2; LT3; LT4; LT5; LT6; LT7; LT8; LT9; LT10; LT11, LT12} where, literals can be represented by linguistic terms as shown in (a). The fuzzy membership function can be defined as in (a) and (b).

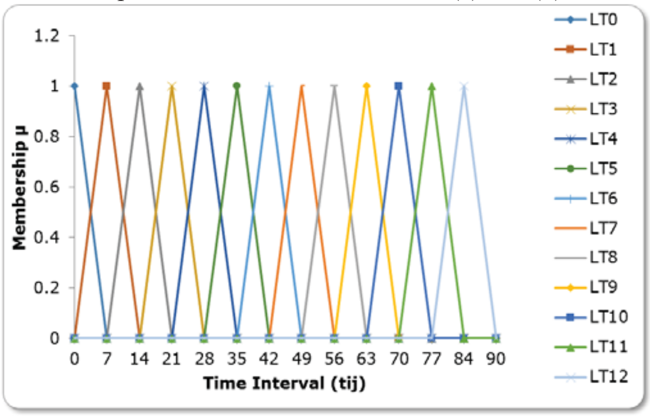



Fig. 4. Membership functions of Fuzzy time interval FTI-13

The graphical representation of the fuzzy membership functions is shown in Figure 4.

Now, let us consider the events with their support count of dataset 2 as in Table I. The optimum fuzzy time intervals are detected on these events by implementing FTI-Apriori-Event algorithm as shown in Table II.

TABLE I EVENT DESCRIPTION- LMS

| Events | Description | Support Count |
|---|---|---|
| 89 | Upload/Download | 98.5% |
| 21 | Attendance | 88% |
| 293 | Evaluation | 80% |

TABLE II OPTIMUM FUZZY TIME INTERVAL BY FTI-APRIORI-EVENT ALGORITHM ON LMS

| Optimal Fuzzy Time Interval for LMS | | | | |
|---|---|---|---|---|
| **Events** | **Support** | **Time Interval** | **Patterns** | **Fuzzy Time Interval (*ltd*)** |
| **Event "Upload/Download" With 99% support** | **80%** | **FTI-3** | 20 | LT1 |
| | | **FTI-6** | 49 | LT1 |
| | | **FTI-13** | 111 | LT1 |
| **Event "Attendance" With 88% support** | **70%** | **FTI-3** | 19 | LT1 |
| | | **FTI-6** | 44 | LT1 |
| | | **FTI-13** | 104 | LT1 |
| **Event "Evaluated" With 80% support** | **50%** | **FTI-3** | 16 | LT1 |
| | | **FTI-6** | 37 | LT1 |
| | | **FTI-13** | 71 | LT0 |

As observed in Table II, for all the time interval partitions such as FTI-3, FTI-6 and FTI-13, an optimum fuzzy time interval is discovered. For event all the events: event "89" (Upload/Download event) with 99%, event "21" (Attendance event) with 88% and event "293" (Evaluated event) with 80% support count, at minimum support of

80%, 70% and 50% respectively, the maximum frequent sequence patterns are found when fuzzy time intervals are partitioned into 13 fuzzy time partitions sets. Hence, the optimum fuzzy time interval detected for future occurrence of event "89" (Upload/Download event) and event "21" (Attendance event) is "LT1" with µLT1 membership function, ranging between 0 and 14 time units or between 0th and 14th day. Similarly, the optimum fuzzy time interval detected for future occurrence of event "293" (Evaluated event) is "LT0" with µLT0 membership function, ranging between 0 and 7 time units or between 0th and 7th day in a 90 days dataset.

FTI-Apriori-Event algorithm not only predicts an optimum fuzzy time for a particular event but also generates reference/sequence list where these optimum time interval event can occur in future as shown in the sequences generated below

*Min-Supp: 80%*

*Upload/Download Event - FTI-13 with time difference of 7 days*

*2-length pattern:*

*89 LT0 89; Fuzzy support: 512.54:2315 Students*
*89 LT1 89; Fuzzy support: 713.21:2382 Students*
*89 LT2 89; Fuzzy support: 597.63: 2332 Students*
*89 LT3 89; Fuzzy support: 471.87: 2294 Students*
*89 LT4 89; Fuzzy support: 378.27: 2289 Students*
*89 LT5 89; Fuzzy support: 335.5: 2385 Students*
*89 LT6 89; Fuzzy support: 308.56: 2200 Students*
*89 LT7 89; Fuzzy support: 273.15: 2290 Students*
*89 LT8 89; Fuzzy support: 210.36: 2290 Students*
*89 LT9 89; Fuzzy support: 134.21: 2193 Students*
*89 LT10 89; Fuzzy support: 68.74: 2250 Students*
*89 LT11 89; Fuzzy support: 21.62: 2209 Students*
*89 LT12 89; Fuzzy support: 2.40: 2127 Students*

The number of sequences generated on execution of FTI-Apriori-Event algorithm on event "89"(Upload/Download) with minimum support of 80% with fuzzy time interval FTI-13 having 7 day time difference, are 111(1- length sequence; 1 in number, 2- length sequence; 13 in number, 3-length sequence; 68 in number, 4-length sequence; 28 in number). For simplicity, the 2-length sequences are shown above. The sequences with the optimum fuzzy time interval detected by the algorithm has the maximum fuzzy support as shown above; example fuzzy linguistic variable LT1. Thus the highest fuzzy support sequences generated by FTI-Apriori-Event algorithm are in the optimum fuzzy time interval detected.

To predict the students list which are at risk and those do not perform the given task in the optimal time detected are generated by taking the compliment set of the sequence list generated for the above case. For example, consider a Set A (2382 students) for simplicity we have considered numbers and not the sequence list or student list which perform the "89" (Upload/Download) event in the optimal time detected "LT1", then the compliment set A' (1-A): (2725-2382=343), will be a set with elements as the sequence list or student list who fail to perform the "89" (Upload/Download) event in the optimal time detected "LT1". Thus predicting the At-Risk students for a particular event or task. This information is very helpful for the instructor and the institution by providing assistance for such students in need; thus increasing the quality and quantity of students in higher education.

## V. Conclusion

Sequential pattern mining [1], is used to discover frequent lists of Web pages ordered by viewing time for predicting visit patterns. In EDM, "Take Action before it's too late", visualize being able to alter student's path if you were alert of risk factors early on. Faculty will be able to help students when they first start showing signs of falling behind or disengaging in their courses. With students browsing patterns or sequence of web click patterns make it easy for faculty to see immediately who's not participating, missing deadlines, and/or performing below average. FTI-Apriori-Event algorithm makes this possible by predicting an optimum fuzzy time interval for the future occurrence of a given event. On execution of FTI-Apriori-Event algorithm, the algorithm not only predicts an optimum time for a particular event but also generates sequence list where these optimum fuzzy time interval sequence can occur in future.

The fuzzy time interval with maximum sequences generated detects the optimum time interval.

In most of the cases, the longest sequences generated by FTI-Apriori-Event algorithm are in the optimum fuzzy time interval detected with maximum fuzzy support count.

FTI-Apriori-Event algorithm not only predicts an optimum fuzzy time for a particular event but also generates sequence list where these events can occur in future with optimum time interval detected.

To predict the students list which are at risk and those do not perform the given task in the optimal time detected are generated by taking the compliment set of the sequence list generated. This information is very helpful for the instructor and the institution by providing assistance for such students in need; thus increasing the quality and quantity of students in higher education.

## References


[1] R. Agrawal and R. Srikant, "Mining sequential patterns", *In Proc. Int. Conf. Data Engineering*, 1995, pp. 3–14.

[2] Y. L. Chen, S. S. Chen, and P. Y. Hsu, "Mining hybrid sequential patterns and sequential rules", *Inf. Syst.*, vol. 27, no. 5, pp. 345–362, 2002.

[3] J. Han and M. Kamber, Data Mining: Concepts and Techniques, New York: Academic, 2001.

[4] Pei, J., Han, J., Pinto, H., Chen, Q., Dayal, U. and Hsu, M.C., "PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth", *Proceedings of International Conference on Data Engineering*, pp. 215–224, 2001.

[5] Han, J., Pei, J., Mortazavi-Asl, B., Chen, Q., Dayal, U., & Hsu, M.-C., "FreeSpan: Frequent pattern-projected sequential pattern mining", *Proceedings of International Conference on Knowledge Discovery and Data Mining*, pp. 355–359, 2000.

[6] Srikant, R. and Agrawal, R., "Mining sequential patterns: Generalizations and performance improvements", *Proceedings of the 5th International Conference on Extending Database Technology*, pp. 3–17, 1996.

[7] Zaki, M. J., "SPADE: An efficient algorithm for mining frequent sequences", volume 42 Issue 1-2, January-February, pp 31–60, 2001

[8] Palit, S.K (1997). Electronic engineering education in India. Proc. 1st Asia-Pacific Forum on Engineering and Tech. Education., Monash University, Melbourne, Australia, p.50-54

[9] National Audit Office (NAO), "Staying the course: the retention of students in higher education," London, 2007.

[10] F. Sarker, H. Davis, and T. Tiropanis, "A Review of Higher Education Challenges and Data Infrastructure Responses," *in proceedings of* International Conference for Education Research and Innovation (ICERI2010), Madrid, Spain, 2010.

[11] Mingming Zhou ;, "Data Mining and Student e-Learning Profiles", 2010 International Conference on E-Business and E-Government (ICEE), pp. 5405 – 5408, 2010.

[12] Patil, Priyanka Anandrao ; Mane, R.V., "Prediction of Students Performance Using Frequent Pattern Tree", 2014 International Conference on Computational Intelligence and Communication Networks (CICN), pp. 1078 – 1082, 2014.

[13] Thomas, E.W., Marr, M.J. ; Thomas, A. ; Hume, R.M. ; Walker, N., "Using discriminant analysis to identify students at risk", Proceedings of Frontiers in Education Conference, 1996. FIE '96. 26th Annual Conference., pp. 185 – 188, 1996.

[14] Schalk, P.D.; Wick, D.P. ; Turner, P.R. ; Ramsdell, M.W., Predictive assessment of student performance for early strategic guidance, Frontiers in Education Conference (FIE), 2011, pp. S2H-1 - S2H-5, 2011.

[15] Singh, M. ; Singh, J. ; Rawal, A., "Feature Extraction Model to Identify At -- Risk Level of Students in Academia", 2014 International Conference on Information Technology (ICIT), pp. 221 – 227, 2014.

[16] Minaei-Bidgoli, B. ; Kashy, D.A. ; Kortemeyer, G. ; Punch, W.F., "Predicting student performance: an application of data mining methods with an educational Web-based system", Frontiers in Education, 2003. FIE 2003 33rd Annual, pp. T2A – 13, 2003.

[17] Krishna Kishore, K.V. ; Venkatramaphanikumar, S. ; Alekhya, S., "Prediction of student academic progression: A case study on Vignan University", 2014 International Conference on Computer Communication and Informatics (ICCCI), pp. 1 – 6, 2014.

[18] Alkhasawneh, R. and Hobson, R., "Modeling student retention in science and engineering disciplines using neural networks", 2011 IEEE Global Engineering Education Conference (EDUCON), pp. 660 – 663, 2011.

[19] Rizvi, M.; Humphries, T. ; Major, D. ; Lauzun, H. ; Jones, M, "A new CS0 course for at-risk majors ",Software Engineering Education and Training (CSEE&T), 2011 24th IEEE-CS Conference on, pp. 314 – 323, 2011.

[20] Sarker, F. , Tiropanis, T. ; Davis, H.C., "Linked data, data mining and external open data for better prediction of at-risk students", 2014 International Conference on Control, Decision and Information Technologies (CoDIT), pp. 652 – 657, 2014.

[21] Wang, X., Ceberio, M. ; Contreras, A.F.G., "Towards fuzzy method for estimating prediction accuracy for discrete inputs, with application to predicting at-risk students", 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), pp. 536 – 539, 2013.

[22] Hai-yan Wu ; Jing-jun Zhu ; Xin-yu Zhang, "The Explore of the Web-Based Learning Environment Base on Web Sequential Pattern Mining", CiSE 2009. International Conference on Computational Intelligence and Software Engineering, 2009., pp. 1 – 6, 2009.

[23] L. A. Zadeh, "Fuzzy sets," Information and Control, vol. 8, pp. 338-353, 1965.

[24] S. Mitra, S. K. Pal, and P. Mitra, "Data mining in soft computing framework: a survey," IEEE Transaction on Neural Networks, vol. 13(1), pp. 3-14, 2002.

[25] J.H Lee and H. L. Kwang, "An extension of association rules using fuzzy sets" in the IFSA, Prague, Czech Republic, pp 399-402, 1997.

[26] T.P. Hong, C.S. Kuo and S.C.Chi, "Mining association rules from quantitative data", Intell. Data Anal., vol. 3, pp. 363-376, 1999.

[27] W. Zhang, "Mining fuzzy quantitative association rules", in Proc. 11th Int. Conf. Tools Artificial Intelligence, Chicago, IL, pp. 99-102, 1999.

[28] G.Chen and Q. Wei, "Fuzzy association rules and the extended mining algorithms", Inf. Sci. vol. 147, pp. 201-228, 2002.

[29] J.W.T.Lee, " An ordinal framework for data mining of fuzzy rules", in Proc. FUZZ IEEE, San Antonio, TX, pp. 399-404, 2000.

[30] C.M.Kuok, A.Fu and M.H.Wong, " Mining Fuzzy association rules in databases", SIGMOD Rec., vol. 27, no. 1, pp. 41-46, 1998.

[31] M.Vazirgiannis, "A classification and relationship extraction scheme for relational databases based on fuzzy logic", in Proc. Research Development Knowledge Discovery Data mining, Melbourne, Australia,pp. 414-416, 1998.

[32] W. H. Au and K. C.C.Chan, "Mining fuzzy association rules", in Proc.6th Int. Conf. Information Knowledge Management, Las Vegas, NV, pp. 209-215, 1997.

[33] W. H. Au and K. C.C.Chan, "An efficient algorithm for discovering fuzzy rules in relational databases", in Proc. IEEE Int. Conf. Fuzzy Systems, vol. II, pp. 1314-1319, 1998.

[34] W. H. Au and K. C.C.Chan, "FARM: A data mining system for discovering fuzzy association rules", in Proc. FUZZ IEEE, vol. 3, pp. 22-25, 1999.

[35] W. H. Au and K. C.C.Chan, " Mining fuzzy association rules in a bank account database", IEEE Trans. Fuzzy Syst., Vol. 11, no. 2, pp. 238-248, 2003.

[36] Yangdong Ye, Qing Zhou , Xu Wang, Limin Jia, "Analysis of fuzzy time interval in the hybrid Petri net model of train operation system", 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), Vol. 1 pp. 644 – 648, 2010.

[37] Chung-I Chang, Hao-En Chueh , Lin, N.P., "Sequential Patterns Mining with Fuzzy Time-Intervals", Sixth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD '09, Vol 3,pp. 165 –169, 2009.

[38] Chung-I Chang, Hao-En Chueh, Yu-Chun Luo, "An integrated sequential patterns mining with fuzzy time-intervals", International Conference on Systems and Informatics (ICSAI), pp. 2294 – 2298, 2012.

[39] Joong Hyuk Chang, "Mining weighted sequential patterns in a sequence database with a time-interval weight", Journal of Knowledge-Based Systems, Vol. 24, issue 1, pp. 1-9, 2011.

# SESSION

# POSTER PAPERS

## Chair(s)

### TBA

# Virtual-C – a Programming Environment for Learning C on Mobile Devices

**D. Schmudde and D. Pawelczak**
Faculty of Electrical Engineering and Computer Science
Universitaet der Bundeswehr Muenchen (UniBw M), Neubiberg, Germany

**Abstract—** *Virtual-C is an integrated development environment (IDE) that was primary developed for desktop systems to fit the needs of programming novices for learning C in undergraduate courses. This paper discusses the port and optimization of the IDE for the usage on mobile devices, especially on tablet computers with Android. Therefore, new fields of application were identified like an example collection for basic programming topics and an online environment for small exercises. A description language based on the JSON syntax was developed for exercise modules. Besides that, the user interface was improved by simplifying the presentation and by adding gesture support.*

**Keywords:** C-programming course, mobile learning, *Qt*

## 1. Introduction

There are different approaches of teaching a programming language in an efficient and graphic way. Mostly a suitable IDE is used to provide features for programming novices and a balanced learning curve [1]. The *Virtual-C* IDE[1] [2] was developed using the *Qt* library from *Qt Project* [3] to reach this goal and concentrates on debugging capabilities, visualizations to show the data and control flow and a mechanism for *HTML*-based exercises during practical courses.

The rising availability of mobile devices in lectures shows the need for an additional orientation of the IDE towards mobile operating systems. Therefore, more users could be reached and new methods of learning can be realized.

## 2. Didactics

The goal is that students can benefit from a mobile version of the IDE and are supported in their individual learning processes. The following use cases describe the main fields of application and how students can take advantage from a port in their study.

### 2.1 Example Collection

The integration of an example collection sorted by topics offers examples for basic programming questions that can be run and debugged to understand the codes' purpose, without the need of writing programs on the mobile device. In addition, examples can be modified directly in the editor to test and analyze alternatives.

[1]https://sites.google.com/site/virtualcide/

### 2.2 Exercises

Programming exercises can help students to deepen their understanding of important programming concepts by leading step by step through small tasks that are collected into exercise modules (EM). After each step the achieved solution is inspected and in case of an error a small hint or a detailed feedback is given depending on the task type. The exercise templates and test files are available through a web server. In that way, the content is not static and can be customized for different courses. The progress is automatically stored on the local device so that even small time periods can be used for continuing exercises.

### 2.3 Editor Mode

The editor mode can be used for any C-programs or projects. Therefore, the most important developing functions are summarized into a menu to simplify the user interface (UI). It can be minimized to provide more space for coding.

## 3. Port and User Interface

### 3.1 Qt Limitations

The benefit from using *Qt* is that the application can be deployed to various platforms including mobile operating systems like *Android* or *iOS*. The drawback of this is that – until now – the support of the library for mobile systems is limited and e.g. the *WebKit* browser that is needed for the original *HTML*-based exercises is not available on mobile devices (compare [2]). For the analysis of alternative views, a description language (DL) was defined to generate *QML* representations of the EMs, see Section 4. By comparing the support on desktop and mobile systems, the presentation quality and flexibility, the handling and sustainability, *Qt Quick 2* showed the best results overall and is therefore the preferred view for exercises and additional content [4].

### 3.2 Gestures and Scalability

The UI of a mobile application has to be flexible to support a wide range of screen sizes and resolutions. This was realized by using different zoom levels and gestures that provide a familiar and easy handling of the IDE.

## 4. Exercise Modules

A DL was defined to combine all necessary information about an EM into a single *JSON* file.

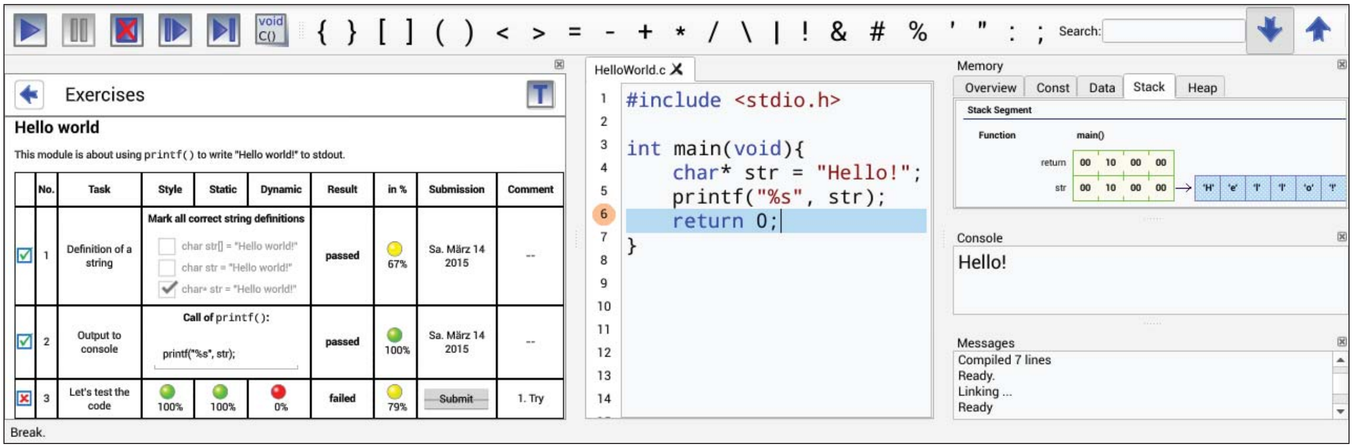



Fig. 1: Screenshot of the IDE running on an *Android* device. The view on the left is used to show the content of exercise modules. It also serves as a menu to provide easy access to the main features like examples, exercises and the editor mode.

### 4.1 Module Description

The module description contains headings in different sizes and other text to introduce the topic and to give necessary information. The text format can be adjusted by using basic *HTML* commands.

### 4.2 Task Types

There are three different task types. The first type initiates a multiple choice request whereas the second type demands a textual user input. After a task submission the input is compared with the solution provided by the EM and the result is calculated as a percentage. The last type needs additional files to test the source code by compiling, linking and executing test functions for a dynamic code behavior analysis. For documentation purposes all results are added into the locally stored EMs.

### 4.3 Exercise Module Example

Fig. 1 above shows a visualization of an interpreted EM described in Fig. 2. It contains a header, a short introduction text and one exercise table including all three task types. (Note that in string values backslash escapes have to be used for double quotes and backslashes itself.)

```
[   { "h1": "Hello world" },
    { "text": "This module is about using <code>printf()
        </code> to write \\\"Hello world!\\\" to stdout." },
    { "exercise": [
        { "choice": {
            "task": "Definition of a string",
            "question": "Mark all correct string definitions",
            "checkbox": {
                "char* str = \\\"Hello world!\\\"": true,
                "char str[] = \\\"Hello world!\\\"": true,
                "char str = \\\"Hello world!\\\"": false
        } },
        { "scanf": {
            "task": "Output to console",
            "question": "Call of <code>printf()</code>:",
            "expected": "The complete call is needed!",
            "solution": "printf\\s*\\(\\s*\"\\s*%s\"\\s*,\\s*
                [a-zA-Z_]+[a-zA-Z0-9_]*\\s*\\)\\s*[;]?"
        } },
        { "check": {
            "task": "Let’s test the code",
            "target": "testHelloWorld.c"
        } }
] } ]
```

Fig. 2: EM example with the DL based on *JSON* syntax.

## 5. Conclusion

The *Virtual-C IDE* focuses on debugging capabilities as well as data and control flow visualizations that are rare on mobile IDEs [4]. With additional content for new fields of application and UI enhancements, the environment is prepared for the usage on mobile devices. Yet, the original version can also take advantages from the port, because the features like examples, exercises and the gesture support can be reused on desktop systems, too. Further, the preparation of EMs or modules for the practical courses can be much faster when using the new *JSON*-based DL. Henceforth, students can practice exercises with the mobile version of the IDE at any time and any place.

## References


[1] E. Dillon, M. Anderson and M. Brown. “Comparing feature assistance between programming environments and their "effect" on novice programmers,” in Journal of Computing Sciences in Colleges, Volume 27 Issue 5, May 2012, Pages 69-77.

[2] D. Pawelczak and A. Baumann. “Virtual-C - a programming environment for teaching C in undergraduate programming courses,” IEEE Global Engineering Education Conference EDUCON, (Istanbul, Turkey, April 3-5, 2014), 1142-1148. Available: https://sites.google.com/site/virtualcide/virtualc.pdf

[3] The Qt website. [Online]. Available: http://www.qt.io/

[4] D. Schmudde, “Port and optimization of the Virtual-C IDE for the usage on mobile devices,” Bachelor thesis, UniBw M (in German: “Portierung und Optimierung der Virtual-C IDE fuer den Einsatz auf mobilen Endgeraeten”), 2014. Available: https://sites.google.com/site/virtualcide/bachelorarbeit.pdf

# SESSION

# TEACHING AND TESTING TOOLS AND METHODS, E-TEXTBOOKS, FRAMEWORKS AND SYSTEMS + ATTENDANCE TRACKING AND RELATED ISSUES

## Chair(s)

**TBA**

# Using QR-Codes for Attendance Tracking

**D. Deugo**[1]
[1]The School of Computer Science, Carleton University, Ottawa, Ontario, Canada

**Abstract –** *Students experiencing difficulty with their studies don't often identify themselves as students in need of support. The Science Student Success Centre at Carleton University actively seeks out these science students to give them the help they require. One indicator of need is a student's grades. Another indicator is attendance in class. In this paper, we describe how we used QR-Codes to help track class attendance. While our existing online card swiping attendance tracking system was suitable for small class sizes, using QR-Codes for attendance tracking proved to be suitable for larger classes.*

**Keywords:** Attendance, Tracking, Android, QR-Codes

## 1 Introduction

In [1, 2], we described the initial view of our attendance tracking system and piloted its usage in classes to examine its performance and usability. Our motivation for building the system came from The Science Student Success Centre (SSSC) at Carleton University. The SSSC collaborates with other on-campus departments to offer workshops, events, and activities that help to develop a student's academic and professional skills. The SSSC also takes an active role in helping science students struggling in their first-year computer science, math, and science courses with the mission as follows:

- Increase the engagement and retention of students in the Faculty of Science
- Foster the growth and achievements of high performing students
- Identify and support students who may be experiencing difficulty with their studies
- Inform students of professional and academic development opportunities
- Aid in the recruitment of outstanding students to the Faculty of Science

One of the questions for the SSSC was how to determine students in need of its services. One indicator it used was to look at first term, mid-term marks in late October. We also believe that looking at students' attendance records for the same period provides another key indicator of students in need. The SSSC already contacts students with marks less than 60% and asks them to come in and visit with the SSSC team to talk about issues they are facing with their courses and suggest possible techniques and actions they can take to overcome the issues. We also felt that looking for students with less than 60% attendance records was another group we should meet with individually and that there would likely be a crossover between low grades and low attendance students.

Taking attendance using paper and pen was one approach we could have used, but we knew it was slow and prone to errors. In addition, the paper method required a data entry phase in order to generate reports, which also suffered from similar problems. Therefore, using pen and paper was ruled out from the beginning. We found in [1, 2] that we could use our student card-based system for tracking attendance in classes containing less than 50 students. What we needed was to expand that capacity.

### 1.1 Goals

Reiterating our goals and objectives from [1]. Our main goal was to provide a fast and efficient attendance tracking system. In addition, the system must work in any and all classrooms at Carleton University, including its electronic classrooms – those with computers and projectors – and those classrooms containing no computers. A further goal was to provide a system that requires minimum hardware, and can be maintained at minimum cost. Our new goal is to break the maximum 50 student class size.

### 1.2 Objectives

To meet the goals, we had the following objectives:

- Use easily found, inexpensive hardware for the system.
- Make use of mobile devices to help with the attendance tracking process but not require the use of student owned mobile devices
- Use open source software to minimize development and maintenance costs
- Installation should be fast and simple
- Attendance reports should provide end users with the greatest flexibility for manipulating the collected attendance data
- Support both Mac and PC platforms, and Android and iOS mobile devices

### 1.3 Outline

In section 2, we further describe literature on existing QR-Code based attendance tracking systems. Section 3 describes our system and how it used QR-Codes in the classes to take attendance. Section 4 describes our timing results. Section 5 provides our conclusion and reviews our system's feature set.

## 2 Background

Several different systems [3, 9] are now using QR Codes as the primary method to track attendance. As [3] mentions, the primary benefit is that:

> *"you can track attendance with no extra expensive bar code readers or similar hardware, using just your smartphone, tablet or PC with webcam"*

Many other systems [3] rely only on smartphones, tablets or PCs running custom applications to manage the entire attendance tracking process, which is often similar in process to the following:

- One creates a user id in order to login to a system manager.
- One creates a class/event to track.
- The system generates a unique URL that one can publish.
- Attendees click the URL and register for one's class or events by entering their name and other data you specify
- Attendees print their QR codes and take them to one's event.
- Attendees scan their QR code at one's event and the system marks them as attended.
- One tracks attendance statistics of one's event by login in to the system.

Pricing models vary, but [3] propose the following:

> *"Pay only for what you use! Buy checkin credits in chunks of 100, 300, 500 or 1000. One credit cost is $0.1 USD. Pay only when your user/student scans the card."*

The benefit of this model is that the attendance data is not stored on the device, but is maintained by a server, allowing for regular backup and maintenance. The disadvantage is that the generate QR Codes are event specific. Therefore, if used in a university setting, students will need to keep and maintain upwards of ten codes a year for all their courses. Another disadvantage to the system proposed by [3] is that it is currently under development and not ready to be used.

One system that is already being used is by [4]. Focused more in the conference and trade-show domain but still available for education requirements, their system creates badges for users that can be scanned using specific equipment located at conference booths or event entrances in order to track attendance or control access. An advantage to this system is that it provides the following capabilities:

> *"generates real-time views on session attendance and duration, exhibit floor visits, traffic patterns and event activities."*

Another benefit is that their system supports QR Codes, Bar Codes, RFID, and NFC. One disadvantage is that their system maintains the collected data on their servers, which in a university setting, may not be possible due to personal information security concerns, policies and laws.

Rather than using a server approach for storing collected attendance data, systems like [5] keep everything on the device and rely on attendance taking to be done manually. As the attendance taker, it is one's responsibility to define the events and people being tracked. There is support for importing contacts, but it is still one's responsibility to add new attendees. The same goes for backup and restoration of attendance data. One uses the application to manually "tick off" people that attend an event as attendees are not required to maintain any form of QR Code.

In [6] the attendance system uses QR codes, a server and proposes that:

> *"the instructor to do nothing extra beyond presenting the slides of the course to the students. Hence, students may register their presence at any time they wish during the class, while having in mind that registration times are recorded. "*

They believe that the time taken by instructors to take attendance may be viewed as a waste of lecture time. Their system relies on the ability of students to perform that attendance tracking activity on their own device. However, relying on students having their own devices and remembering to use the corresponding application on them can be problematic in certain settings. If attendance tracking is important, we believe there is a strong reliance on the involvement of the instructor. Moreover, in classes using experiential learning [7], where learning is through reflection on doing, the one-on-one interaction between student and instructor during class permits attendance taking to be done at that time, as part of the interaction, rather than before or after class where no real interaction is done.

In [8], QR Codes are used for checking attendance rather than taking attendance. With this approach, instructors track attendance using Excel spreadsheets that are forwarded for processing and entry into a database by an administrator.

Instructors post QR codes on class doors or select points where students can scan them using their phones in order to review their absentee rate. The benefits of the system are that it eliminates the need for a notice board identifying students' absentee rates and helps keep student data private.

Focused in the conference domain, where check-ins and session tracking is important, [9] uses a combination of QR codes that each attendee have, typically on their conference badge, and conference support staff having iOS and Android devices running their custom application to scan and track attendees. Their system provides real time reports and registration details to registered users. The disadvantage of this system is that it focuses in the domain of conferences, where the number of events and attendees is usually small. A mid-size university can have hundreds of classes and over 25000 students to track on a daily bases. The sizes of the education domain make typical costing models around attendees and sessions expensive.

While none of these approaches are adequate for the educational domain, they do suggest several important features:

- Real time reports
- Fast Scanning
- QR Code per attendee
- User rather than attendee device app
- End User id and registration

All of these features are provided in our approach described in the next section

# 3 Approach

As indicated in [1, 2] in our system users are instructors or people that are taking attendance in classes or at different events. They are provided with a user id, which is their email address, and a unique password. After logging into the system using the Event Tracker website, shown in Figure 1, users can download their attendance real time report for any class or event.

In the previous versions of our system, users attached a USB card reader and swiped student cards in order to track attendance. We also had a mobile app that students could use on their phones, to identify themselves as attending the class or event. In the current version of our system, these capabilities are still supported. However, we have added the ability for students to generated custom QR codes that are scanned by users or instructors using a custom application that runs on a smartphone or tablet. The scanned information is then sent back to the system's server and a person's attendance at a class or event is record. These steps are described in more detail in the following sections.

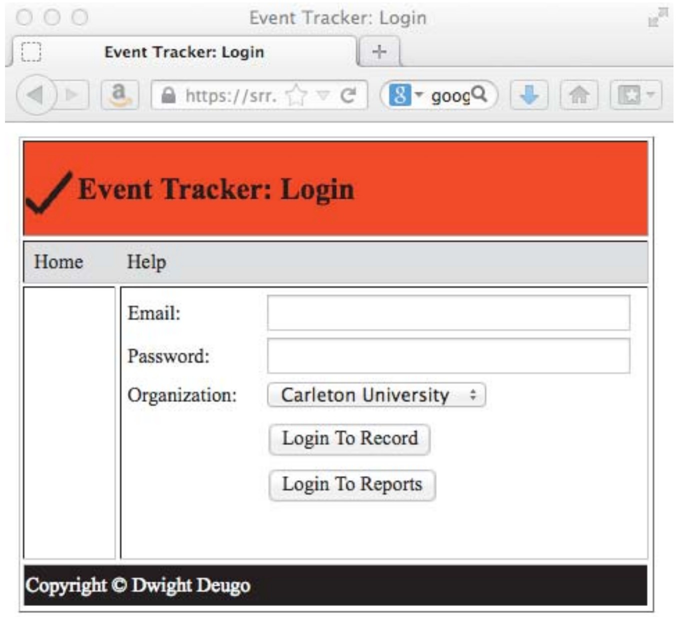



Figure 1: Login Screen

## 3.1 QR Code Generation

Students generate their QR Codes by going to our QR Code Generation website, as shown in Figure 2. At the site their first step is to enter their relevant information: first name, last name, student id, and email address. For general events, there is also a comment field included to collect other relevant data. The second step is for students to click the Generate Your QR Code button. This will generate a QR code on the web page as well as display the information in the QR Code. Using any scanner app, such as the Android QR Barcode Scanner app, one can immediately scan the generated QR Code. Figure 3 is the result of scanning the code in Figure 2 by the QR Barcode Scanner app on an Android device.

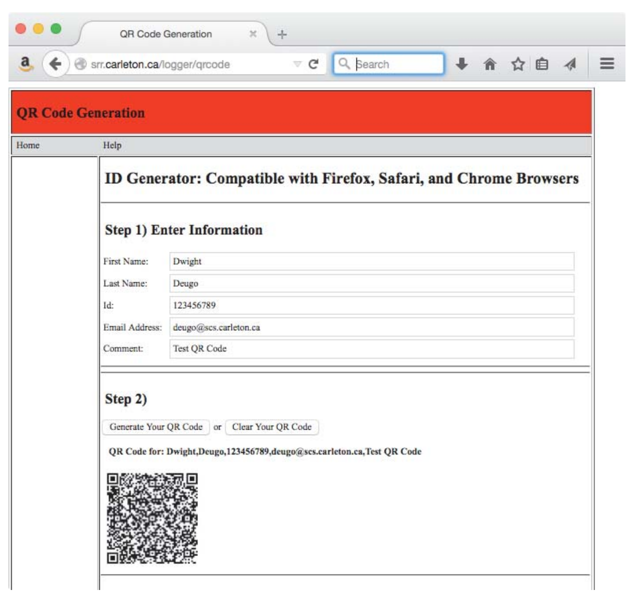



Figure 2: QR Code Generator


Figure 3: Standard Barcode Scanner Result

Step 3 in the process at the website, shown in Figure 4, is for a student to save their QR Code. A student has three options. The first, step 3.1, is to hit the Print Your QR Code button that will display the QR code on its own web page and initiate the corresponding browser's printing functionality. One can print the QR Code on a printer or print it as a PDF file that can be saved to hard disk. By using step 3.2, a new web page is created containing the QR Code, as shown in Figure 5. Using the browser's File->Save Page As… menu option the QR code can be saved as a GIF file with a given name. Using this option the QR Code can also then be printed, emailed, or moved onto a student's mobile device.

Provided a student enters the correct Recaptcha text, hitting the Email Your QR Code To Above Email Address, step 3.3, will do as the name suggests.

The QR Code Generation and its three steps provide a quick and easy approach for student to generated their individualized QR Codes and be able to save them to disk or print them out. The next part of our approach is for students to bring their QR Code to class in order for their attendance to be tracked.

## 3.2 Taking Attendance

Taking attendance using the QR codes is done through a mobile app running on an Android device. The single view app, shown in Figure 6, named EScanner, permits users to login, scan and upload their collected attendance tracking data.

Before logging in, a user can configure the server IP address, how scans are recorded (manual or automatic) and can clear out local scans saved for bulk upload by accessing the app's settings, as shown in Figure 8.

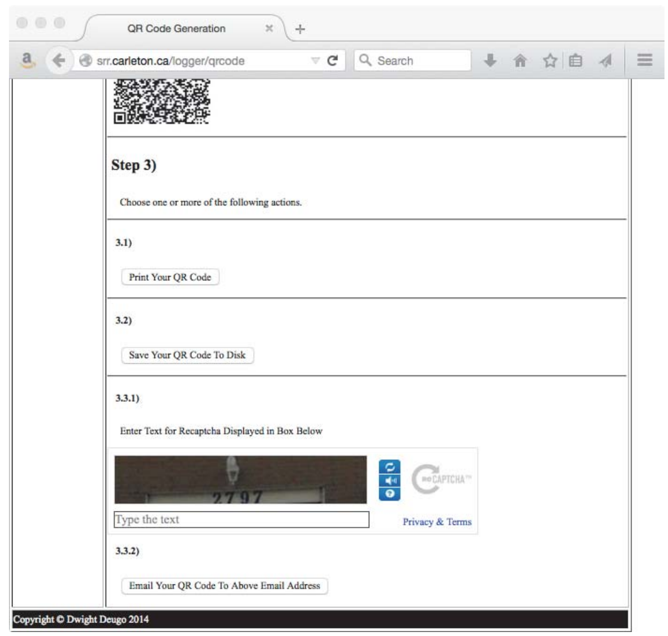



Figure 4: Saving QR Codes

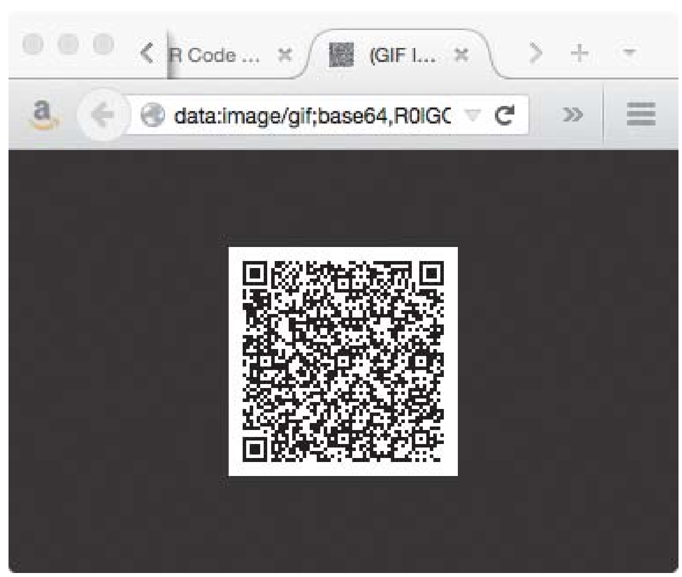



Figure 5: QR Code

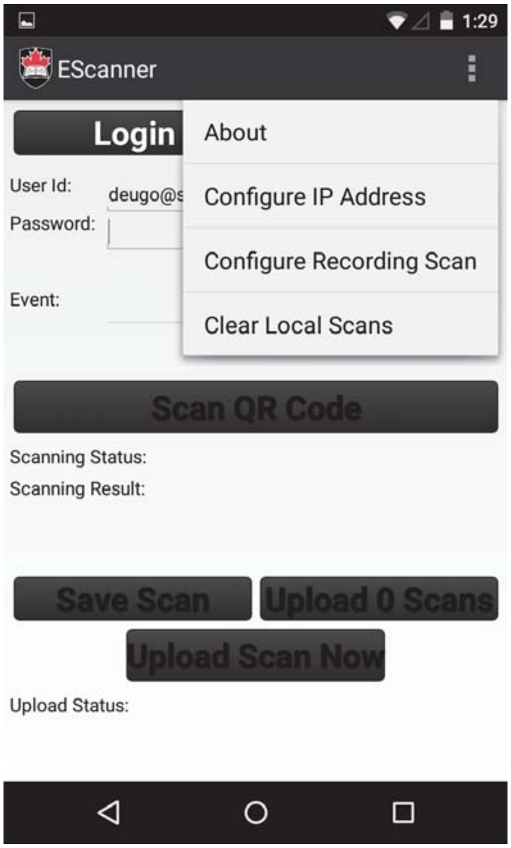



Figure 6: Settings

An instructor who has been given an account id and password simply enters that information and selects the Login User button. If the login is successful, the screen changes to what is shown in Figure 9. Choosing the About options provides the user with a quick look at their existing settings, as shown in Figure 7.


Figure 7: Settings Overview

After a successful login, the user is presented with a list of courses/events they are responsible for. After selecting the correct event, the user is ready to scan an attendee's QR Code. By selecting the Scan QR Code button the user is transferred to their device's camera application where they can simple point it at a students QR code and it will capture the information for the code. Figure 9 shows the results of scanning the QR Code shown in Figure 5. Provided the code is on of our custom codes, the user can immediately upload the scanning data to the server, or save it on the device for uploading later. The delayed upload feature is useful when the device is not connected to the network. The application can currently store up to 1000 scans for uploading later. Whether the scans are uploaded immediately or saved locally depends on which of the corresponding buttons the users selects after scanning the QR Code. However, in the options menu the user can turn on automatic uploading or automatic saving to speed up the process of scanning QR Codes with out having to select the completion action.

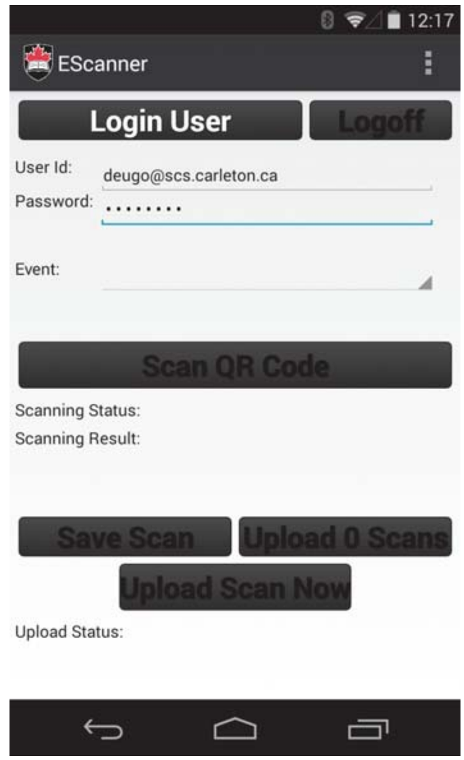



Figure 8: EScanner

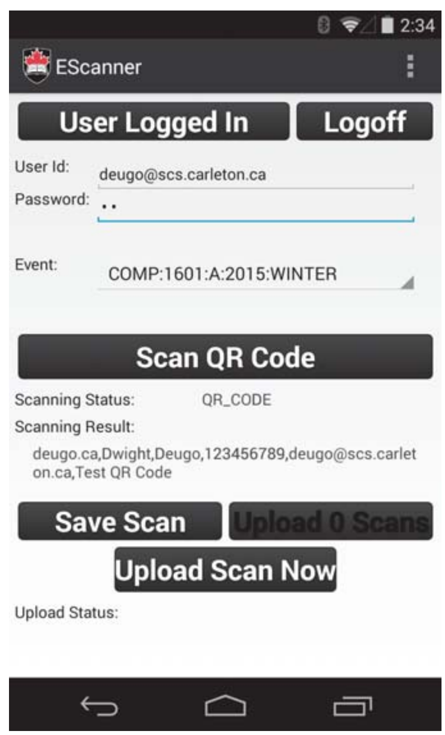



Figure 9: After Scanning

Once a user logs out of the app, saved scanning data remains intact but no uploading of saved data or new scans are possible.

# 4 Results

Of significance to us is how quickly it takes to perform individual attendance tracking? This question can be broken down into different parts. The first part is how long does it take to scan one person? The second part is how long does it take to send the scanned information to the server. If a user selects automatic upload, how long does the combination of scanning and uploading take? The final question is, if the user simply scans and chooses to do a bulk upload of scans, how long does this action take.

To answer the first question, we used our mobile application to scan the QR Codes of 10 people 10 times and recorded the total time required to perform the scannings. The average and median results for scanning 10 QR Codes are displayed in Table 1. Next we uploaded 10 scan results one by one to the server and computed both the average and median times to send the individual result, shown in Table 1. The third row in Table 1 identifies how long it takes to scan and upload a QR Code. This happens when a user of the mobile application has configured the corresponding combined option. This option results in one less button push for the user to actually upload the scan results. The final row in Table 1 indicates how long it takes to bulk upload 10 scan results. Again the average and median results are presented.

Table 1: Scanning and Uploading Timings

| | Average Time (sec) | Median Time (sec) |
|---|---|---|
| Scanning of 10 QR Codes | 50 or 5 per scan | 50 or 5 per scan |
| Uploading of 1 QR Code Scan Result | 2.3 | 2 |
| Scanning and Uploading of 1 QR Code Scan Result | 5.7 | 6 |
| Bulk Upload of 10 QR Code Scan Results | 3 or 0.6 per scan result | 3 or 0.6 per scan result |

From these results we can compute the following: The time required to take 1 QR Code scan and then upload it is (5 + 2.3) 7.3 seconds on average or (5 + 2) 7 seconds on median. If we chose to do automatic uploading immediately after scanning, the total times are decreased to 5.7 and 6 seconds respectively. However the fasted results are achieved when we scan many people and then do a bulk upload of those results. Using this option for 10 people, it took (50 + 3) 53 seconds to perform the task or 5.3 seconds per person.

Then end result is that performing scans and then doing a bulk upload would let us track, under the best conditions, approximate (600 / 5.3) 113 students in a ten minute period (600 seconds) before the start of a class or event. The best conditions would be were the queue of attendees with QR Codes is large enough where there is always another code ready to be scanned after the previous scan and there are no time delays connecting to the server to perform the bulk upload.

# 5 Conclusion

We believe our current mobile app and server-based system can be used for tracking attendance in classes of up to 100 people. Moreover, if students attend different events, they do not need register with subsequent events, as the system does not require an attendee to have a different QR Code. Our custom QR Codes contains all of the required attendee information. And, unlike the previous versions of our system, there is no time required to register an attendee with the system before they are tracked.

After working on three different versions of our system, we see the following feature set emerging as the basis for different attendance tracking system:

- Tracking Data Location: The location of event tracking data is usually stored either on the device, on a server, or a combination of the two. In this last case, data is stored locally on a device for later uploading to a server.
- Attendee Identification: attendees of event tracking systems identify themselves with cards, QR Codes, id numbers, or something on a device/card that is unique. In some systems, the id/card/code is verified before attendance is tracked. In other systems this task is not a requirement.
- Tracking devices: tracking devices range from a few devices running a mobile application that is used by one or a few users to track the attendance of a single event to systems where every attendee has a device with a mobile app they used to indicate their attendance at an event.

A final issue, which in itself is a feature, is the technique to ensure that an attendee is actually at an event when being tracked. If every attendee has access to a required mobile application, then the potential is there for them to identify themselves as at and event when they are not actually there. Using the location of the device is one technique one could use to help determines a device's location, but depending one when the device starts, stops, has Wi-Fi, cellular, or GPS information, the identified location can be inaccurate. With QR Code scanning, students must be at the event or class.

Our working system has the following features. Ultimately, attendance tracking data is stored on our server. We support the creation of a user with corresponding events they wish to track attendance. Users can login to our system with their user id and password in order to retrieve real time Excel reports of who has been tracked at one of their events. We have a web site where those attending events generate a unique, custom QR Code that they bring with them to any class or event. The custom QR Code only works with our system, and can be printed on paper, or stored as an image on a laptop or mobile device. Only a user, not an attendee, has access to the mobile app required to track attendance. This fact ensures that attendees are at actually at the event, as their codes are scanned at the same location. Finally, while attendance tracking data is stored on our server, we support the having a user scan up to 1000 QR Codes and storing them on their device before bulk uploading them. The reason for this feature is two fold: The first is that it increases the speed at which attendees can be scanned and processed at an event. Secondly, if the mobile device is having difficulty connecting to the network, the user can still process attendees and upload the scanning data at a later time.

# 6 References

[1] Deugo D. “Attendance Tracking”. The 9th International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS 2013), 79-85. 2013.

[2] Deugo, D. Tracking Attendance Pilot. The 10th International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS 2014), 101-107. 2014.

[3] QR Bubble – attendance tracking. Web 15 May. 2015. <http://www.qr-attendance.com/>.

[4] Attendance Tracking Software. Web 15 May. 2015. <http://www.alliancetech.com/solutions/attendee-tracking/intelligent-attendance/overview>.

[5] Attendance Tracker – Android. Web 15 May. 2015. <https://play.google.com/store/apps/details?id=peterman.apps.attendance&hl=en>.

[6] Masalha, f. and Hirzallah, N. A Students Attendance System Using QR Code. (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 5, No. 3. 2014.

[7] Corbett, A.C. Experiential Learning Within the Process of Opportunity Identification and Exploitation. Entrepreneurship Theory and Practice, Volume 29, Issue 4, 473–491. July 2005.

[8] Baban, M.H.M. Attendance Checking System Using Quick Response Code for Students at the University of Sulaimaniyah. Journal of mathematics and computer science, 10 (2014), 189-198. 2014.

[9] Attendance & Session Tracking. Web 15 May. 2015. <http://www.eventrebels.com/products/er-mobile-onsite>.

# Gakuzai: Editable Electronic Textbook Education System

**Atsuya Takedani[1], Kousei Ueta[1], Takumi Yamaguchi[1], Takahiko Mendori[2], Yusuke Nishiuchi[1], Masanobu Yoshida[1], Hironobu Satoh[1], Haruya Shiba[1]**

[1] Dept. of Electrical Engineering and Information Science, National Institute of Technology, Kochi College, 200-1 Monobe-Otsu, Nankoku-city, Kochi, Japan

[2] School of Information, Kochi University of Technology, 185 Miyanokuchi, Tosayamda, Ka-mi-city, Kochi, Japan

**Abstract** - *We have been developing a next-generation editable electronic textbook education system called Gakuzai. In the Gakuzai system, an e-textbook is provided to teachers and students. In the Gakuzai system, user can removes (electrically hidden) the unimportant part of the long description of the electronic textbook although leaving only the important part. Teachers will be able to make in a short time the documentation for the class; students can easily create their own notebook.*

*A trial lesson was conducted using the Gakuzai system. One half of the students used paper textbooks and paper notebooks, while the other students used the Gakuzai system. At the end of the lesson, all students answered a quiz and a questionnaire. The average scores for each group were compared by Welch's t-test. No significant difference was observed. The results of the questionnaire indicate that student academic motivation improved by using the Gakuzai system.*

**Keywords:** ICT-based learning, e-textbook, reduction edit, learning style

## 1 Introduction

Education styles in East Asian countries, such as Japan and South Korea, commonly make use of blackboards. Using a blackboard, teachers describe textbook content and write important words, while students copy the blackboard content to their notebooks. This is one-way knowledge transfer from the teacher to the student. The disadvantage of this style is the significant effort required by the students. In some East Asian countries, national projects are attempting to improve education using information and communication technology (ICT) and related electronic equipment in the classroom [1], and researchers have evaluated the effects on education of these projects [2], [3]. However, it is difficult to improve education dramatically by simply computerizing the learning environment.

To significantly improve education, new education-al styles that incorporate electronic technology are considered necessary. In addition, active learning has been attracting attention as a technique that can improve education. However, any changes to education must consider older generation teachers who may not be aware of modern education techniques, such as active learning.

We propose the Gakuzai system, which provides a new educational style based on ICT [4], [5]. With the Gakuzai system, an editable electronic textbook (e-textbook) is provided to teachers and students. When the student develops understanding of a teacher's explanation, they can hide the corresponding content in the e-textbook (leaving only the important terms). As a result, the e-textbook is rewritten to note that only important terms remained for students. Because this work time is shorter than to copy blackboard content, the student is expected to quickly acquire knowledge. The e-textbook is then rewritten using an electronic operation, and the understood content can be obscured.

During class time (e.g., during a lecture), important words or equations can be marked up electronically in the e-textbook. These functions can be useful when students review or do homework. Using the e-textbook, notes can be taken more easily than with current methods (i.e., paper notebooks). In addition, the e-textbook facilitates more effective memorization and solving of problem exercises. As a result, education can be improved. Note that the Gakuzai system can also use electronic blackboards and can support traditional educational styles.

Two web applications have been developed for the Gakuzai system. One application is used to edit the e-textbook, and the other is used for self-learning based on edited e-textbook data. The Gakuzai system provides an environment that supports all study processes, including teacher led classwork and homework. The e-textbook con-tent is written in HTML, and the Gakuzai system has been developed using JavaScript, PHP, and a MySQL database. Therefore, the proposed system can be used with any ICT platform.

This paper describes the Gakuzai system and, through evaluation of a trial lesson, discusses the effects the proposed system has on learning.

## 2 Related Work

There are many cases to evaluate the educational effects of education using ICT.

Piccoli, Ahmad and Ives ware compared the effectiveness of a web-based virtual learning environment with a traditional classroom [6]. Their results indicate that, in the context of IT basic skills training in undergraduate education, there are no significant differences in performance between students enrolled in the two environments.

Bonham, Deardorff and Beichner ware compared web based homework with paper based homework [7]. "Comparison of student's performances on regular exams, conceptual exams, quizzes, laboratory, and homework showed no significant differences between groups." They conclude the change in medium itself has limited effect on student learning.

Sun, Flores and Tanguma ware examined the relevant experiences of college students in terms of how the use of e-textbooks may enhance their learning [8]. Their results indicate that "e-textbooks are perceived as enhancers of student learning experiences in two complementing routes: (1) E-textbook Helpfulness enhances students' learning outcomes directly and (2) Student Involvement plays an important mediating role between E-textbook Helpfulness and Learning Outcome if students use e-textbooks in class."

It has cited technical issues of problems and the e-textbook of student motivation in learning using ICT. Shaw and Marlow have been reported when it becomes negative attitude "When the student likes traditional learning-style and does not accord with past experience" [9].

To solve the problems that have been pointed out in the studies described above, it is important to raise the motivation of the learner. The Gakuzai system is aimed at motivate by reducing the burden of the shortened learner and teacher working time in class.

## 3 Gakuzai System Overview

The Gakuzai system has the following functions.

A) The e- textbook can be quickly edited during lessons.
B) The content can be reproduced.
C) The content is learnable at home.
D) The system must facilitate collaborative learning.

With these functions, the students can hide the corresponding content in the e-textbook (leaving only the important terms). The e-textbook is rewritten to note that only important terms remained for students. Since these operations can be quickly, the note-taking time is shorter than the conventional note take. As a result, students can focus on description of class. The image is shown in Fig.1.

In the creation of time, more detailed explanation, simple experiment, can be performed with high quality learning such as the time to solve the exercises. It will lead to the improvement of the learning efficiency.

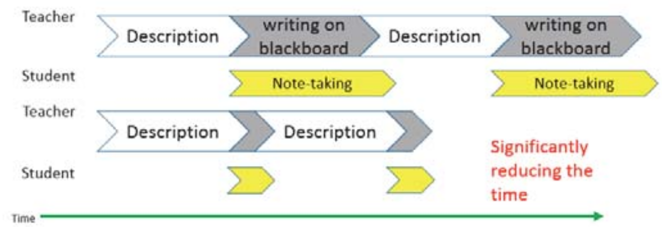



Fig. 1. Screen shots of the Gakuzai system: An improvement image of time shortening

Note that individual and group study can be supported using these functions. The Gakuzai system has adopted HTML that has been adopted because it is software (e.g., operating system) and device independent, and can be easily customized. The Gakuzai system includes a server that distributes the e-textbooks and maintains edited e-textbook data. The Gakuzai server was developed using PHP and Java-Script. Screen shots of the Gakuzai system are shown in Fig. 2. In Fig. 2, the left and right images show e-textbook content from before and after a lesson, respectively. In the right (after) image, a long description, which was in the upper part of the left (before) image, has been replaced by a short sentence.

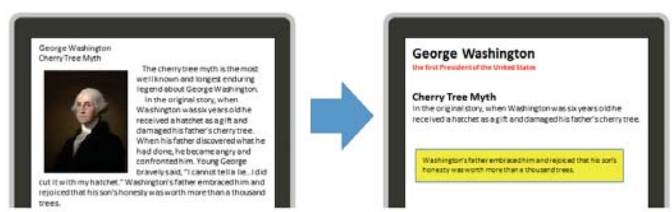



Fig. 2. *Screen shots of the Gakuzai system: e-textbook content before (left) and after (right) a lesson [e-textbook content taken from MOUNT VERNON: Cherry Tree Myth].*

### 3.1 E-textbook editing and marking functions

The Gakuzai system has the following editing and marking features mainly.

A) Unnecessary parts can be hidden or replaced using a short phrase or single word or one blank character. It can also be undone by clicking. Screen shots of the Gakuzai system are shown in Fig. 3.

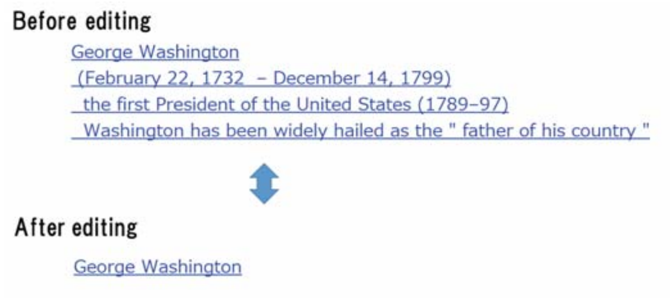



Fig. 3. *Screen shots of the Gakuzai system: Replace function.*

B) Important words can be marked (e.g., text color, underline, and highlights). This feature can be used in place of a coloring by the conventional marking pen. Screen shot of the Gakuzai system are shown in Fig. 4.

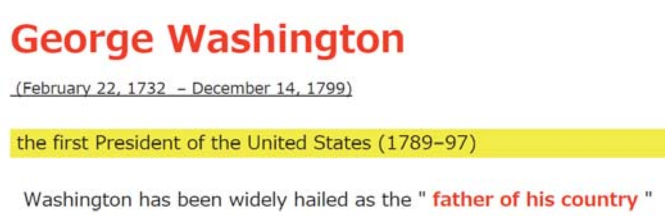



Fig. 4. *Screen shot of the Gakuzai system: Making function to the important words and phrases.*

C) Popup comments can be associated with e-textbook content. Thus, it can hereby supplement the part. Screen shot of the Gakuzai system are shown in Fig. 5.


Fig. 5. *Screen shot of the Gakuzai system: Popup function.*

## 3.2 Self-learning support functions

This function learner developed with the goal of providing support for individual training. It can perform the making of the issue of summary and practice of the notebook electronically. The self-learning support function of the Gakuzai system has the following functions.

A) For memorization-based learning function

The Gakuzai system has an individual study function (Fig. 6). Using this function, a student can mark (text color, underline, bold type, and highlight) the e-textbook content. Marked words are automatically collected using a program and can be used to make formulas collection or important terms collection. In addition, marked words can be used for individual study; these words can be hidden in the e-textbook to facilitate rote (memorization-based) learning.

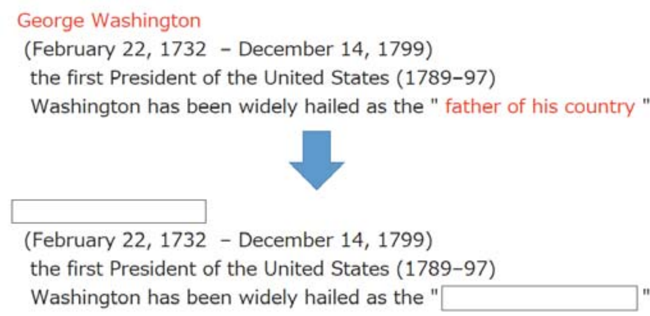



Fig. 6. *Marking e-textbook content: text color changes (upper); marked words are hidden for rote learning (lower).*

B) For practice exercise function

This function can hereby practice the meaning of the important words and phrases and word.

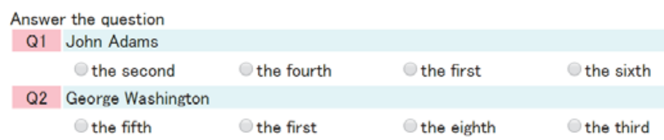
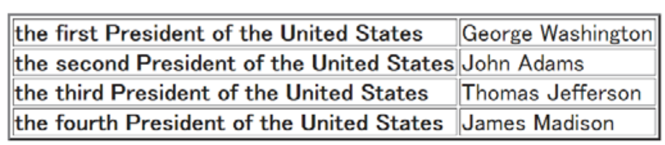



Fig. 7. *Screen shots of the Gakuzai system: The multiple-alternative question making function.*

C) The extraction function of important words

From the content of the e-textbook that students were marked or replaced, it is a function to create a list to pick up important terms**.** The student can expect knowledge fixation by learning with this list repeatedly. Since had been handmade such list in the conventional learning method, it is possible to greatly shorten the time, and anytime remake the list.



| | |
|---|---|
| the first President of the United States | George Washington |
| the second President of the United States | John Adams |
| the third President of the United States | Thomas Jefferson |
| the fourth President of the United States | James Madison |

Fig. 8. *Screen shots of the Gakuzai system: The extraction function of important words.*

# 4 Difference with the conventional learning

We show in the following comparison with the conventional class.

## 4.1 Student side

Describing the student flow of learning techniques of when using the Gakuzai system.

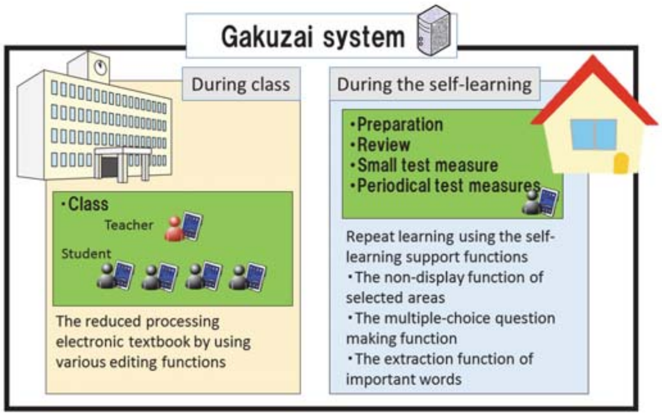



Fig. 9. *A flow of education / the learning using the Gakuzai system*

A) Preparation phase

The student perform a self-learning in the edit function on the basis of the e-textbook.

B) Class phase

The student edits to reference his e-textbook. The e-textbook of the student becomes the notebook only for oneself through this work.

C) Review phase

The student reviews it with one's own notebook that he edited by a class. And he edits it again

D) Small test measure phase

The student studies in reference to one's own notebook that he gathered up. And the students practice it repeatedly by a self-learning support function.

E) Periodical test measures phase

The student studies in reference to the document of the small test measure and one's own notebook.

### 4.2 Teacher side

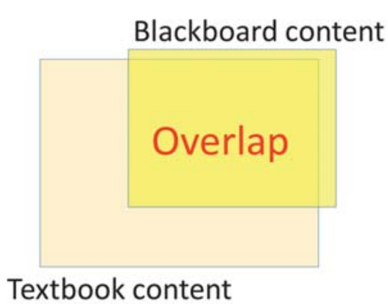



Fig. 10. *Overlap image*

The contents that a teacher writes on a blackboard is contained the contents on a textbook. That image show in Fig. 10. When a teacher uses Gakuzai system, the teachers can easily create a material for use in the classroom, leaving only the important part of the e-textbook.

## 5 Trial Lesson and Evaluation

### 5.1 Trial Lesson and average quiz score

A trial lesson was given to students of (42 students; approximately 17 years old). The trial lesson was conducted using the proposed Gakuzai system. One half of the students used paper textbooks and notebooks, while the other half used the Gakuzai system. Fifty min were allotted for the trial lesson. The first 10 min were used to de-scribe the functionality and purpose of the Gakuzai system. The following 30 min were used to deliver a lesson that described the discharge phenomenon. The final 10 min were used to conduct a quiz on the discharge phenomenon and complete a questionnaire about the proposed system. The average scores of the quiz for each group were compared using Welch's t-test. No significant difference was observed (paper notebook: 4.81; Gakuzai system: 4.29).

### 5.2 Questionnaire survey

A five-scale questionnaire was administered to evaluate the functionality and usability of the proposed system, where (5 is the highest and 1 is the lowest rank). The questionnaire included the following questions.

Q.1: Would you like to use the Gakuzai system?

Q.2: Do you want to try a learning method that we propose?

Q.3: Do you think the Gakuzai system could improve learning efficiency?

Only students who used the Gakuzai system replied to the following.

Q.4: Was the Gakuzai system easy to use?

Q.5: Do you think the Gakuzai system is convenient?

Q.6: Should we improve the Gakuzai system?

Q.7: Please write a free impression.

TABLE 1: Survey Results (values in parentheses are answers from students who did not use the Gakuzai system)

| Quantity | 5 | 4 | 3 | 2 | 1 | Average |
|---|---|---|---|---|---|---|
| Q.1 | 14 (7) | 17 (8) | 6 (4) | 3 (1) | 2 (1) | 3.90 (3.90) |
| Q.2 | 14 (8) | 14 (4) | 10 (8) | 3 (1) | 1 (0) | 3.88 (3.90) |
| Q.3 | 5 (3) | 15 (5) | 11 (8) | 9 (4) | 2 (1) | 3.29 (3.24) |
| Q.4 | 0 | 3 | 7 | 8 | 3 | 2.48 |
| Q.5 | 2 | 8 | 6 | 4 | 1 | 3.29 |
| Q.6 | 14 | 5 | 2 | 0 | 0 | 4.57 |

The results of the questionnaire are shown in Table 1. Note that the values in parentheses are answers from students who did not use the Gakuzai system. The average value for the answers to Q.4 is 2.48, Q.5 is 3.29, and Q.6 is 4.57. Many students considered that the system had poor usability and that it should be improved. However, the average values for Q.1 and Q.2 are very high; they are expected that this system improve their academic achievement. Free impressions comment are 21, comments were 12 on Improvement of usability. Cause of the low average score of the quiz for using the system group is the operation of the Gakuzai system has been difficult. To examine the effect of the Gakuzai system must be longer period observed.

# 6 Conclusion and Future Work

We have developed the e-textbook Gakuzai system. Using the Gakuzai system, students can edit and mark an e-textbook. An initial trial lesson was conducted using the proposed Gakuzai system, and the average scores of a quiz administered to students who used the proposed system and traditional notebooks were compared. No significant difference was observed. In a questionnaire administered to the participating students, despite the low degree of completion of the proposed system, student would expect to academic ability improvement of the Gakuzai system, but it was not found that academic ability could be improved by using the Gakuzai system. In future, we plan to improve the usability of the Gakuzai system and administer additional lessons to facilitate further evaluation.

# Acknowledgments

We would like to thank the students of the National Institute of Technology, Kochi College E2 class who have cooperated in trial lesson.

This work was supported by JSPS KAKENHI Grant Number 25330431.

# References

[1] Japan Ministry of Internal Affairs and Communications: Future School Promotion Research Society.

http://www.soumu.go.jp/main_sosiki/joho_tsusin/eng/Releases/Telecommunications/110720_a.html

[2] A. J. Rockinson-Szapkiw, J. Courduff and et al. “Electronic versus traditional print textbooks: A comparison study on the influence of university student's learning”; Computers & Education, Vol. 63, pp. 259–266 (2013).

[3] N. Matsuuchi, Y. Nishiuchi and et al. “TERAKOYA Learning System Linking Multipoint Remote Users for Supplementary Lessons”; The Journal of Information and Systems in Education, Vol. 9, pp. 35–44 (2011).

[4] H. Shiba, K. Ueta and et al. “Implementation of a Learning Style by E-Textbook Contents Reduction Processing”; HCI International 2013 - Posters’ Extended Abstracts, Communications in Computer and Information Science, Vol. 374, pp. 94–97 (2013).

[5] H. Shiba, K. Ueta and et al. “Self-educate Function Added on Gakuzai System”; HCI International 2014 - Posters’ Extended Abstracts, Communications in Computer and Information Science, Vol. 435, pp. 145–149 (2014).

[6] Piccoli, Ahmad and et al. “Web-Based Virtual Learning Environments: A Research Framekwork and a Preliminary Assessment of Effectiveness in Basic IT Skills Training”; MIS Quarterly, Vol. 25, pp. 401–426 (2001).

[7] Bonham and Deardorff et al. “Comparison of student performance using web and paper-based homework in college-level physics”; Journal of Research in Science Teaching, vol.40, pp.1050-1071 (2003)

[8] Sun and Flores et al. “E-Textbooks and Students’ Learning Experiences”; Decision Sciences Journal of Innovative Education, vol.10, pp.63-77 (2012)

[9] Shaw and Marlow. “The role of student learning styles, gender, attitudes and perceptions on information and communication technology assisted learning”; Computers & Education, vol.33, pp.223-234 (1999)

# A new Testing Framework for C-Programming Exercises and Online-Assessments

**Dieter Pawelczak, Andrea Baumann, and David Schmudde**
Faculty of Electrical Engineering and Computer Science,
Universitaet der Bundeswehr Muenchen (UniBw M),
Neubiberg, Germany

**Abstract -** *Difficulties with learning a programming language are wide spread in engineering education. The use of a single integrated programming environment for coding, debugging, automated testing and online assessment lowers the initial burdens for novice programmers. We have developed the Virtual-C IDE especially for learning and teaching the C programming language with an integrated framework for program visualizations, programming exercises and online assessments. A new enhancement of the IDE is a xUnit like testing framework allowing on the one hand larger sets of small, test-based programming exercises and on the other hand simplifying the development of programming assignments. The integration of the new testing framework in the assessment system gives students a better and direct feedback on their programming achievements and helps to find syntactic and semantic errors in their source code.*

**Keywords:** C-programming, teaching programming, unit testing, static code analysis, dynamic code analysis

## 1 Introduction

Difficulties with learning a programming language are a well-known challenge for students and lecturers in undergraduate courses [1]. As several studies show, continuously practicing programming by starting from small problems shows respectable success, e.g. see [2]. For small classes, training might be part of the lectures and additional tutors can give a hand to prevent students from falling behind. Although the same could be done for large classes, training during the lecture becomes less effective due to the high diversity of previous knowledge of the students, and finding sufficient and appropriate tutors is an extensive task. Luckily an advantage of learning programming is that – after managing an initial burden – students can directly grasp, what happens by debugging or testing their programs. A direct integration of testing in an adequate IDE further lowers the burdens for novice programmers. Ideally students work continuously on small programming exercises and receive directly feedback from the IDE with respect to syntactical and semantic issues; and finally students can submit larger programming assignments from that IDE to receive their credit points.

We have developed the *Virtual-C IDE*[1] (especially designed for learning and teaching the C programming language) over the last years [3]. We use the IDE for automatic assessment and grading of programming assignments in the third year now. However the original aim to have many small accompanying programming exercises for self-learning could not be established yet, due to the high effort for writing tests. In this paper we present a new testing framework, which enormously reduces the effort for test development. Although this framework allows students to write their own tests, we do not plan to integrate test writing in the primer C programming course at the moment, as our curriculum covers software testing in the software engineering courses in the major terms.

## 2 Review of related work

Software testing is a core topic in computer science. Even though software testing is often taught in conjunction with software engineering, it becomes more and more important for programming courses: besides testing first approaches, tool-based testing is widely used today in programming primers [4]. The benefit of testing for students is obvious: with test programs provided by the lecturer, students can train programming outside classroom and have immediate feedback on their exercises. For the Java programming language, jUnit tests are widely spread. The language independent testing concept is typically named *x*Unit tests [5]. Based on these, systems for automated grading of programming assignments like e.g. *AutoGrader* have evolved [6]. As *x*Unit testing is aimed more at professional developers, several tools to simplify test specifications or handling of tests have been introduced for teaching programming, as for instance *Web-CAT* [7]. While *Web-CAT* additionally focuses on tests written by students, other platforms analyze beyond unit testing the data and control flow of programs, like e.g. *ProgTest* [8]. *AutoGradeMe* works independent from unit testing and is based on static code analysis and flow analysis of Java programs [9].

An important aspect is the actual purpose of testing: while assessment tools typically test, evaluate and grade a program source *after* its submission, *x*Unit tests provide immediate feedback without formal grading. The new testing framework presented in this paper covers both: an *x*Unit like test system for offline training and evaluation of programming exercises as well as an automated assessment system for programming assignments. As the test framework is directly integrated in the IDE, students benefit from a single environment for coding, debugging and testing.

---

[1] https://sites.google.com/site/virtualcide/

# 3 Testing framework

The testing framework (TF) is generally based on the well-known *x*Unit frameworks and its test dialogs [5]. It is syntactically adapted from the *Google C++ Testing Framework* [10] with regards to the C programming language and for educational scope. A test suite (TS) is a single test file and consists of test cases (TC) based on one or more tests. Main differences (despite the programming language) compared to the *Google C++ Testing Framework* are:

- Random test data via the macro `ARGR`.
- Simplified verification of output parameters with the `ARG` macro.
- Predefined tests: function and reference function tests (Section 3.4.2), performance tests (Section 3.4.3) and I/O tests (Section 3.4.4)
- No test fixture macro `TEST_F`. Instead, test fixtures are provides in the test prologue.
- Test and test case names can be string literals
- Heap and data segment re-initialization per test case for full application tests, i.e. execution of `main()`.
- Dynamic re-linking of C functions (Section 3.4.5)
- Automatic prototyping for functions under test.
- GUI based selection/ de-selection of test cases.

## 3.1 Constraints for the educational scope

It is a challenge for automated testing of students' source codes on the one hand to support students in finding and fixing of programming mistakes in their code, and on the other hand not to reveal too much information about the solution. The use of a reference implementation inhibits disclosure of the test definition. Still a reference implementation is a common and efficient way to test students' code, compare e.g. [6] [8]. Revealing the test definition (as it is self-evident in software engineering) can help students in understanding the task description and train their testing skills, see e.g. [4]. The TF in general allows to open and edit test definitions except for automatic assessment of programming assignments, i.e. for graded submissions. Theoretically a student can adapt her/ his code according to the test results, thus creating a solution, which fits to the tests but not to the task description. Likewise undesirable is, that students add workarounds in their existing solution to fit the tests. This raises the question about the level of detail for the test results. Our former approach was to give a programming description with detailed specification on the functions and the program's I/O. The tests performed for the assessment ran locally in the *Virtual-C IDE* with direct feedback. However the test input was not revealed in the test results. Although most errors could be fixed easily comparing the test report with the exercise description, student's missed the lack of test data. The new testing framework offers the opportunity to reveal test data in the test results as we introduced random data. This highly increases the number of tests and hardens students to program according to the test results instead of the specification. To check on specific failures, the student can deselect tests that already passed to focus on his/ her errors.

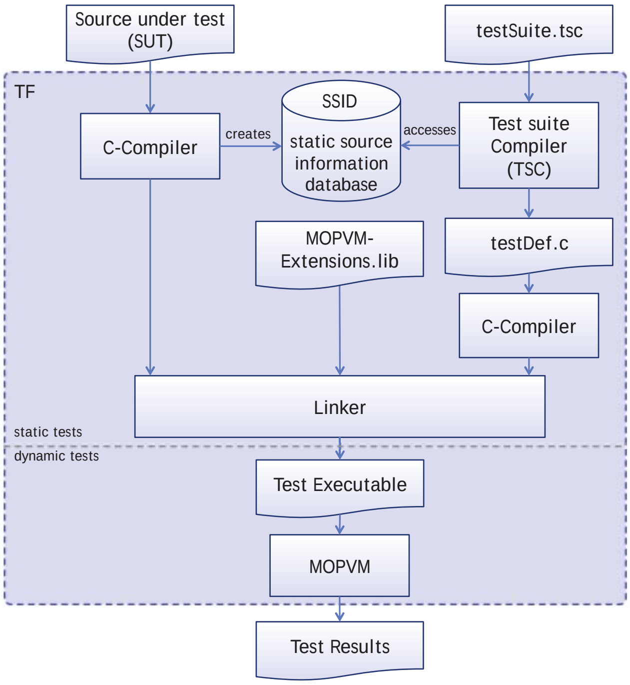



Figure 1. Structure of the testing framework (TF)

## 3.2 Structure of the testing framework

Figure 1 shows the structure of the TF. The first step covers compiling the student's source file under test (SUT). In case of a compiler error, the test execution is aborted, as a syntactical correct file is expected. Static source code information is stored on a function base in a database (SSID). Static information covers for instance the parameters, number of integer or float-operations, the maximum loop depth, recursion, etc. Afterwards the test suite is compiled by the test suite compiler (TSC), which generates a C file of the corresponding test cases. This file is compiled and linked with the SUT and together with the virtual machine (MOPVM) extensions library. Finally the test is executed in the MOPVM and the results are displayed in the test dialog, see Section 4.1. The TSC is a stand-alone tool, which is able to generate test files according to a test suite description; still it relies on the SSID and the MOPVM features integrated in the *Virtual-C IDE*.

## 3.3 Static tests

The compiler issues typical warnings during its semantic analysis phase with respect to typecasts, unused or un-initialized local variables, dead-code, accessing NULL-pointers, etc. Another important static test is performed by the TSC, as it checks, if functions under test (FUT) are properly used in the SUT, e.g. if count and types of arguments are correct. This measure prevents linker errors, which are typically difficult to trace for students. The results of the static tests are printed to the console and visualized in the test dialog, see Section 4.1.

## 3.4 Dynamic tests

Dynamic tests are the main focus of the framework. The test suite can run multiple test cases on a function basis as well as on a program basis. In both cases, a test fixture is set up. Instead of mock objects as specified by the Google C++ Testing Framework [10], the TF provides per test case a *test prologue* and *test epilogue*, as you can see in the following examples Figure 2-6. The test fixture is defined in the prologue; in between prologue and epilogue each test will use a clean test fixture. Optionally TCs can share local variables between tests. For each test case, even the data and heap segments are restored. Consecutive tests inside a test case share data and heap segments, which is for instance important when testing a set of functions with respect to linked lists. The epilogue allows modifying the overall test case result, adding text to the test report or performing clean-ups as, e.g. freeing system resources. The test is run and results can be evaluated afterwards (black-box testing) or during the execution (white-box testing). Black-box testing is preliminary done by evaluating return values, output parameters or console output on given parameters or console input. White-box testing can be achieved by function injection: the linker uses dynamic linking in test mode; thus every function can be re-linked during run-time to a test, a mock or a test-and-mock function, see Section 3.4.5.

### 3.4.1 Assertions vs. expectations and warnings

In accordance with the Google C++ Testing Framework [10] the TF distinguishes between assertions and expectations as expressed by the macros `ASSERT_*` and `EXPECT_*`. An assertion must be met and contradiction leads to an immediate failure of a test. An expectation might not be fulfilled. This will lead to a failure of the test, but its execution is continued. A typical example for expectation vs. assertion is a function modifying a pointer passed as parameter. It is wrong, if the student does not test for a NULL pointer; still the functional part might be implemented correctly for valid pointers. In case the programming assignment does not rely on the NULL pointer test, this test could use `EXPECT_*`, whereas the proper functionality is tested by assertions. The behavior of assertions and expectations can be expressed with the macros `FATAL()` and `ERROR()` respectively, to print a corresponding message in the report. Additionally, a test can always print warnings with the macro `WARN()`.

### 3.4.2 Function tests

In addition to the `TEST()` macro as specified in [10], the TF defines the two macros `_funcRefTest()` and `_funcTest()`. Both macros allow a simple but powerful notation for function tests; the first requires a reference implementation for comparing the results. This short test description is possible by implicitly invoking assertions for given parameters and return values and by adding functional extensions to C. For every function test the TSC uses reflection by querying the SSID for function return and parameter types. Figure 2 shows an implementation of a simple TC including four test descriptions. The `_funcRefTest()` macro expects the name of the FUT, a corresponding reference function, a factor specifying the count of allowed instructions compared to the reference function and the arguments for the function call. The `ARGR()` macro generates random test data in a given range for a specified type. Per default, each `ARGR()` adds three tests (additive, not combinatorial); an optional fourth argument can specify the number of tests. Thus the `_funcRefTest()` example in Figure 2 actually creates six tests. For pointers a pointer to the specified value is created. Strings are treated different, as `char` or `wchar_t` pointers are commonly used for character arrays; thus `ARGR()` creates a modifiable null-terminated string array of printable ASCII characters with a length corresponding to the given range (actually the allocated memory will always refer to the maximum range, only the string length varies). In a function tests with `_funcTest()` you provide the number of allowed instructions together with a list of arguments. For functions, the last parameter is an expression of the expected return value, compare Figure 2. This macro can be easily used to test fix values or to forgo a reference implementation.

```
/* a reference function */
int refMax(int a, int b) {
    return a > b ? a : b;
}

_testPrologue("Maximum test",       // name for the report
    { int x = 17; }                 // a dummy test setup
);

/* a function test with a reference function */
_funcRefTest(max,                   // function under test (FUT)
    refMax,                         // reference function
    5,                              // 5 times more instructions
                                    // are allowed for execution
    ARGR(int, INT_MIN, INT_MAX),    // random argument a
    ARGR(int, INT_MIN, INT_MAX)     // random argument b
);

/* a function test with a calculated result */
_funcTest(max,                      // FUT
   0,                               // default instruction limit
   a = ARGR(int, INT_MIN, x, 1),    // random argument a
   b = rand()%x,                    // random argument b
   a > b ? a : b                    // expected result
);

/* a function test with a fixed result */
_funcTest(max,                      // FUT
   0,                               // default instruction limit
   -10,                             // argument a
   -15,                             // argument b
   -10                              // expected result -10 not 0!
);

/* a google test, $name refers to the test case */
TEST($name, testZero) {
    ASSERT_EQ(max(0,0),0);          // expect return zero
}
_testEpilogue();
```

Figure 2. Function test definitions

```
/* a simple reference procedure */
void refStrAppend(char* x, const char* append) {
    strcat(x, append);
}

/* a test case with variable s as test setup */
_testPrologue("Append string test", {
    char s[128] = "Hello "; }
);

/* a function test with reference function */
_funcRefTest(strAppend, refStrAppend, 5,
            ARG(char*, s, 128), "World" );

/* same test with ARG parameter */
_funcTest(strAppend, 0, ARG(char*, s, 128, "Hello World"),
    "World");

/* tests with strings of random size */
_funcRefTest(strAppend, refStrAppend, 5,
    ARG(char*, s, 128),    /* first argument s */
    ARGR(char*, 0, 120)); /* a random string  */

_testEpilogue();
```

Figure 3. Tests for output parameters

Function output parameters can be tested with the `ARG()` macro. In case a non-constant pointer parameter is passed via the macro, the result is compared with the argument of the reference implementation or the optional forth argument of `ARG()`; e.g. `ARG(char*, s, 128, ”Hello World”)` checks, if the contents of `s` is “Hello World” after the function call. The third parameter defines the maximum allocated memory size. Figure 3 shows a test case with three different simple tests on strings. The second test uses the `ARG()` macro to feed an in-/ output parameter and to verify its contents afterwards. The third test uses `ARG()` in combination with a reference function.

```
/* performance on insertion in a binary tree */
_testPrologue("Insert in binary tree", // name
    {},                             // empty clean test fixture
    { tBinTree *root = NULL; } // shared test fixture
);

TEST($name, insertMike) { insert(&root, "Mike"); }
TEST($name, insertFred) { insert(&root, "Fred"); }
TEST($name, insertAnn)  { insert(&root, "Ann");  }
TEST($name, insertStan) { insert(&root, "Stan"); }
TEST($name, insertRose) { insert(&root, "Rose"); }

TEST($name, performanceLeaf) {
    double ratio = (double)$insertAnn / $insertRose;
    if (ratio < 0.99 || ratio > 1.01)
        FATAL("Ann & Rose inserted at different cost!");
}
TEST($name, performanceTree)
{
    if ($1==$2 || $1==$3 || $1==$4 || $1==$5)
        FATAL("Insertion of root/node at same cost!");
    if ($2==$4 || $3==$5)
        FATAL("Insertion of node/leaf at same cost!");
}
_testEpilogue();
```

Figure 4. Performance tests (insertion in binary tree)

### 3.4.3 Performance tests

Performance tests evaluate the number of instructions required for the execution of a FUT; the instruction counter can be queried with the MOPVM extension library function `_getExecutionCount()`. Each tests initially resets the counter, so that the counter can be evaluated in a `TEST()` macro. To access the execution counter from other tests within a TC, the instruction counter is additionally stored in the pseudo variables $1 … $n for n test cases. So each test can compare the performance of the previous tests. The execution count of a test can also be queried by $*testName*, as long as the test name is specified as a regular identifier, compare e.g. `$insertAnn` in Figure 4. These variables can be evaluated either in a `TEST()` macro or in the epilogue. Figure 4 shows a simple and far not complete test case checking on the performance of a binary tree insertion. The test case expects, that insertion of leafs at the same depth require about the same count of instructions. The insertion of the root, nodes or leafs in different depth cannot be performed with the same count of instructions, as an insertion in an array for instance would allow.

### 3.4.4 I/O tests

A console C program typically reads data from *stdin* and prints results to *stdout*. I/O tests can be performed on functions or whole programs. The MOPVM extensions library allows simple redirection of *stdin* and *stdout*. The `_IOTest` macro requires a string literal as input for *stdin*. Instead of the NUL-character, EOF is passed to the application. The optional third and further arguments present *stdout*. This is a list of string literals representing a regular expression on the expected or (with the !-Operator) unexpected output plus a descriptive error message. Alternatively, the test can have a body for an explicit test definition: the pseudo variable `$return` refers to the return value of the FUT whereas `$out` can be used to check on *stdout*, compare Figure 5.

```
/* an I/O test case for Fibonacci numbers */
_testPrologue("Fibbonacci output");

/* simple positive I/O test */
_IOTest(main,                 // call main()
        "5\n8\n10\n",         // input is 5 8 10 plus enter
        "\\b5\\b",            // regular expression, output
        "\\b21\\b",           // should contain 5, 21 and
        "\\b55\\b",           // 55 in that order.
        "Wrong output for the input 5, 8 and 10" //error
);

/* simple negative I/O test */
_IOTest(main, "11\n",         // input is 11 plus enter
        !"55",                // output should not contain 55
        "For input 11 your program should not print 55!"
        );

/* explicit checks on stdout */
_IOTest(main,                // call main()
        "3\n14\n-1\n"        // input is 3 14 -1 plus enter
) {
    if ($return == 0)        // check return value of main
        WARN("main() should return EXIT_FAILURE for -1.");
    if(!_containsRegEx("\\b2\\b[^3]*377\\s", $out))
        FATAL("Fibonnaci of 3 and 14 expected!");
    if (!_containsRegEx("error|invalid|illegal", $out))
        FATAL("Error message expected for -1");
}
_testEpilogue();
```

Figure 5. I/O tests

```
#include <stdarg.h>
/* a test case for the use of scanf & format specifiers */
int scanfCalls = 0;

int myscanf(const char*format, ...) {
    va_list argptr;
    va_start(argptr,format);

    switch (scanfCalls) {
        case 0: if (!strstr(format,"%hhc"))
            ERROR ("use the specifier %hhc at first");
            break;
        case 1: if (!strstr(format,"%lf" ))
            ERROR ("use the specifier %lf second");
            break;
        default:
            break;
    }
    scanfCalls++;
    return vfscanf(stdin, format, argptr);
}


_testPrologue("Format specifiers",{
    _relinkSymbol(scanf, myscanf); // call myscanf instead of scanf
    }
);

_IOTest(ReadFromInput(),"a .4711") {

    if (scanfCalls<2)
        FATAL("You function ReadFromInput should call scanf twice!");
}
_testEpilogue();
```

Figure 6. Function injection

### 3.4.5 Function injection

The *Virtual-C IDE* uses dynamic linking for testing, i.e. the VM maintains a look-up-table for each function. A test case can modify the look-up-table with the `_relink-Symbol()` function by overwriting the original function pointer with a function pointer to a mock or a test function.

This allows replacing any function as long as the new function provides the same signature. Figure 6 shows a test case on the `scanf()` function by replacing `scanf()` with `myscanf()`. This function counts the number of calls as well as it checks the format specifiers. The function injecttion is done here in the test fixture, thus it is active throughout the test case. Function injection can also be done on a test basis, i.e. each test can provide it's own mock function. The original function linking is restored when running the next test case, thus the following test case will operate again on the original `scanf()`-function unless it is re-linked again.

# 4 Field of application

The testing framework has two major modes of operation. By opening test suites (file extension .tsc) the test dialog is opened and tests can be directly run in the IDE. This mode is called the exercise mode, as it is designed for self-learning. The same dialog can be indirectly opened as part of the automated assessment system (assessment mode).

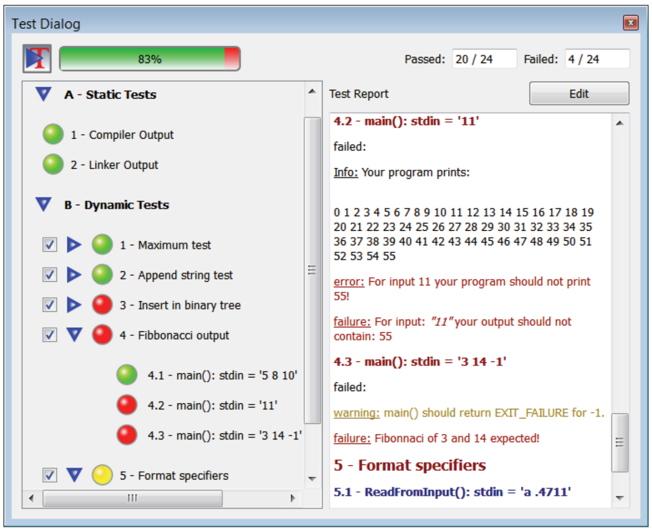



Figure 7. Test dialog of the exercise mode (EM)

## 4.1 Exercise mode

A student can open a test file and run it on her/ his C-module to receive a report on her/ his achievements. Figure 7 shows an example dialog of the exercise mode (EM). The results of the static and dynamic tests are directly visualized in a traffic-light scheme: red (fatal failures), yellow (errors), green (pass). The test report is printed to the console. In addition to *x*Unit tests, the user can directly deselect test cases or select specific test cases in the dialog to focus on single failures. The test code is initially hidden, to focus on the testing; the "edit" button allows viewing and editing the test specification, as described in Section 3.4. EM is for self-learning or lecture accompanying; the lecturer can provide an exercise description together with a test file to allow students testing their solutions. As the IDE supports direct opening from URLs, a test file can also be published on a web server.

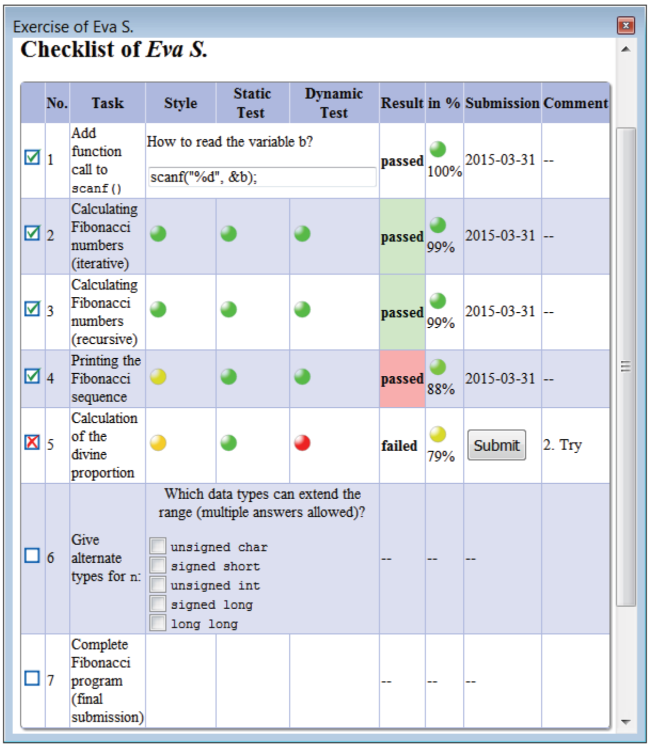



Figure 8. Submission dialog of the assessment mode (AM)

## 4.2 Assessment Mode

In assessment mode (AM) a student works on a larger programming assignment. She/ he has to submit her/ his solution from the *Virtual-C IDE* to a server. The student's code is checked and the progress is stored on the server. Unless the code has passed the tests, the student can submit corrections of his/ her code. After a successful submission, the student can continue locally in EM to enhance her/ his solution. Programming assignments can consist of multiple tests. Figure 8 shows an example of the submission dialog, which is a plug-in of the Virtual-C IDE. The dialog is implemented as a web view and controls the workflow of a programming assignment. The submission dialog is actually presented as a questionnaire with embedded test suites – for details see [3]. It is an html document presenting the tasks and gathering the results. Each submission is stored on the server. For a code submission, the test suite will be downloaded from the server and executed as if the test is executed in EM. Afterwards the results are updated in the submission dialog:

- *Style*: coding style warnings – additional style checks
- *Static Test*: test results from static tests (EM)
- *Dynamic Test*.: test results from dynamic tests (EM)
- *Result*: an overall percentage based on the criteria above

A threshold is defined for each criterion, so that too many compiler, style or linker warnings might already abort further tests. In case a test fails, the EM dialog is opened to present the detailed test results. In opposite to EM, the student cannot continuously run tests, as the number of test runs is limited to prevent try-and-error submissions. The student is not allowed to edit or view the test suite, as it may contain a reference implementation. In addition to EM, style checks and plagiarism detection are performed.

#### 4.2.1 Coding Style

Today's software developer tools widely support auto formatting of source code. Nevertheless, the authors think, that following code styling rules – especially with respect to structuring source code by proper indentation – is still a competence students should achieve in programming courses. The IDE therefore does not provide auto format but checks the source code against a set of rules like for instance:

- Indentation; a consistent indentation is expected throughout the code: either K&R style or ANSI style. Proper indention in conditions and loop bodies.
- Identifier names are using the proper upper/ lower case conventions for variables, functions, defines, etc.
- No use of magic numbers.

The coding style test is a build-in function of the *Virtual-C IDE*. In AM the style test is mandatory, i.e. a source code submission without any indentation (as for instance received via email) won't pass the style test.

#### 4.2.2 Plagiarism Detection

It is beyond doubt, that plagiarism detection is required in any automated assessment system with respect to source code. As plagiarism is an academic offence, it is handled completely different compared to programming faults; as an option, a plagiarizing student can be blocked from any further submissions to trigger a talk with the course instructor. After the talk, the student can continue working on his/ her assignment if applicable, as the course instructor can enable re-submission. Results from the plagiarism detection are presented with a traffic-light background color in the result report, but are not included in the overall percentage. In Figure 8 the first two program submissions have passed the plagiarism detection and are marked with a green background, whereas the last submission failed. For details on the plagiarism detection system see [11].

### 4.3 Offline Mode

A third mode is the offline mode (OM), which is nearly identical to AM. It allows performing a programming assignment offline, i.e. either from a local repository or from a webserver but with unidirectional access. OM can serve as a preparation for the examination or to provide additional more extensive exercises. OM is also important for the lecturers or course instructors to prepare the test suites of a programming assignment.

## 5 Evaluation

The automatic assessment system is used in our C programming course for three years now. Students prepare their programming assignments at home and are allowed to submit their code during two hours class time. Each programming assignment typically consists of five consecutive code submissions. During class time 2-3 instructors are present to support 4 groups of about 20 students each. Initially the system was installed to reduce the administrative work of the instructors, to reduce plagiarizing and to focus more on programming issues.

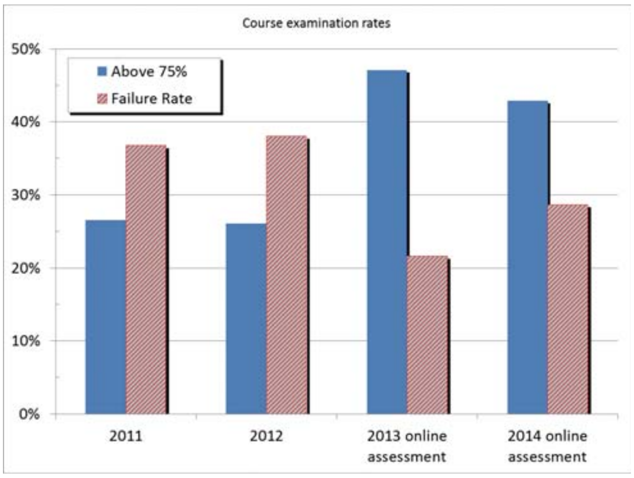



Figure 9. Course examination results compared with online assessment from 2013 and without before

### 5.1 Code submissions

The formal functional tests of the automated assessment system require, that students put more time into their programming assignments with respect to fixing errors compared to the years before. In addition, each submission is treated equally; instructors cannot turn a blind eye to minor programming mistakes or to collaborative work. This had a positive effect on the examination results: students were more comfortable with typical programming constructs, compare Figure 9.

### 5.2 Test reports

Properly explaining a failure to a student is the most difficult part of the automated assessment system. The advantage of the *x*Unit based testing is, that assertions are printed in the report in a "standardized" format, like, e.g.: *expected 55, but result is 34*. The TSC will additionally add the corresponding function call for function tests in the report, like e.g. *fibonacci(10)*. Students have the most difficulties with exceptions. Although the test report prints the line in the source code, that is responsible for the exception, students hardly find the problem on their own. Reasons for exceptions were tests, passing NULL pointers, as well as erroneous code like uninitialized pointers, unallocated or too less allocated memory and array bounds. Our expectation was that students debug their failing functions by simply copying the function call from the test report to their source code. But with respect to function testing, students seem to be overextended. They often seem to flinch from changing their code for debugging. For I/O tests on the opposite, students usually run their programs with the test input without difficulties.

### 5.3 But-it-works syndrome

As other studies show, students perform rarely tests on their code with a high coverage [12] [13]. So a failure in their submission is often taken as an error of the test system or harassment. Unfortunately not all students read the assignment description properly. They might for instance print the Fibonacci number to *stdout* inside the function `fibonacci()` instead of returning the calculated number as requested. The program gives the expected output to screen, but the function test self, of course, fails. Another typical fault is storing the return value in a global variable instead of using the return statement; and again the function test will fail. Although these examples can be easily explained to a good student, as they represent unstructured programming habits, other students often see the modification of a working solution just as additional work. Laborious but effective is to add a reference to the assignment description in the report, e.g. *your function does not return the expected value as described in section … on page …*

### 5.4 Test development

Writing tests with the new testing framework is exceptional easier compared to writing tests by directly using the function based MOPVM extension library (compare [3]). Especially beneficial is the automated report generation and the simplified random data generation. Thus an instructor must put less time in developing tests; still the test coverage is much higher as the number of actual tests rises due to the simple randomization of arguments.

## 6 Conclusion and outlook

The new testing framework integrated in the *Virtual-C IDE* enables students to develop, debug and tests their programs in a single programming environment. Small test suites provided by the course lecturer can serve as accompanying exercises with little effort for the lecturer. At the same time, the test framework smoothly integrates into an automated assessment system. We expanded the system towards a better reporting, an appealing visualization and higher test coverage. In opposite to secret tests for programming submissions, details on the test data is laid open to students in order to give a better feedback for fixing errors.

Although the TF supports performance tests there is still a high potential in pushing performance tests further. A lack of the assessment system is, that code fitting the requirements will mostly pass even if it is written cumbersome or less effective. So good students may miss an opportunity to discuss their solutions with the course instructors or fellow students because of a failure or an unexpected poor feedback. An additional report on the quality of the submission could trigger such a discussion for the benefit of these students. An ongoing research at our institute is detailed analyzing the dynamic structure of programs, which should result in a metrics for code quality.

## 7 References

[1] A. Robins, J. Rountree and N. Rountree. Learning and teaching programming: A review and discussion. Computer Science Education, Vol. 13, (2003), 137-172

[2] W. Pullan, S. Drew and S. Tucker. A Problem Based Approach to Teaching Programming. In *Proc. of the 9th Int. Conf. on Frontiers in Education: Computer Science and Computer Engineering* (Las Vegas, USA), FECS’13, 403-408

[3] D. Pawelczak and A. Baumann. Virtual-C - a programming environment for teaching C in undergraduate programming courses. In *Proc. of IEEE Global Engineering Education Conference EDUCON*, (Istanbul, Turkey, April 3-5, 2014), 1142-1148

[4] J. L. Whalley and A. Philpott. A unit testing approach to building novice programmers’ skills and confidence. In *Proc. of the 13th Australasian Computing Education Conf.* (Perth, Australia), ACE 2011, 113-118

[5] K. Beck, Test Driven Development: By Example. Addison-Wesley, 2002.

[6] M. T. Helmick. Interface-based programming assignments and automatic grading of Java programs. In *Proc. of the 12th annual SIGCSE conf. on Innovation and technology in computer science education* (Dundee, Scotland, UK, June 23-27, 2007), ITiCSE’07, 63-67

[7] S. H. Edwards. Using software testing to move students from trial-and-error to reflection-in-action. In *Proc. of the 35th SIGCSE technical symp. on Comp. science education* (Norfolk, Virginia, USA, March 3-7, 2004) SIGCSE '04, 26-30.

[8] D.M. de Souza, J.C. Maldonado and E.F. Barbosa. ProgTest: An environment for the submission and evaluation of programming assignments based on testing activities. In *Proc. of 24th IEEE-CS Conf. on Software Engineering Education and Training* (Waikiki, Honolulu, HI, USA, 22-24 May, 2011), CSEE&T, 1-10

[9] D. M. Zimmerman, J. R. Kiniry and F. Fairmichael. Toward Instant Gradeification. In *Proc. of 24th IEEE-CS Conf. on Software Engineering Education and Training* (Waikiki, Honolulu, HI, USA, 22-24 May, 2011), CSEE&T, 406-410

[10] Zhanyong Wan, et al. Google C++ Testing Framework – googletest: https://code.google.com/p/googletest/.

[11] D. Pawelczak: Online Detection of Source-code Plagiarism in Undergraduate Programming Courses. In *Proc. of the 9th Int. Conf. on Frontiers in Education: Computer Science and Computer Engineering* (Las Vegas, USA), FECS’13, 57-63

[12] Gómez-Martín, M. A., and Gómez-Martín, P. P. Fighting against the 'But it works!' syndrome. In *Proc. XI Int. Symp. on Computers in Education* (Coimbra, Portugal, November 18-20, 2009). SIIE’09.

[13] S. H. Edwards and Z. Shams. Do student programmers all tend to write the same software tests? In *Proc. of the 2014 conf. on Innovation & technology in computer science education* (Uppsala, Sweden, June 21-25, 2014), ITiCSE '14, 171-176

# Generation of Complexity-Controlled Mathematics Problems for Colleges

**Yuki Mori**[1] **and Shuichi Yukita**[2]
[1]Graduate School of Computer and Information Sciences , Hosei University, Tokyo, Japan
[2]Faculty of Computer and Information Sciences, Hosei University, Tokyo, Japan

**Abstract**— *Exercises with adequate complexity motivate students and facilitates a deeper understanding. Manually constructing such problems consumes time that teachers can otherwise use to mentor students. Many software tools and services for automatic generation of mathematics problems are available on the web, but they provide only materials up to high school level. In addition, no standardized methods are provided to evaluate and control the computational complexity of generated problems. In this paper, we proposed a framework for evaluating computational complexity from the leaners' perspective, aiming to apply our framework to the automatic generation of college-level mathematics problems with controlled computational complexity. Our framework helps teachers prepare learning materials and thereby save time for mentoring students.*

**Keywords:** Linear Algebra, Computational complexity, Hermitian matrix, Unitary matrix

## 1. Introduction

We propose a framework for evaluating the computational complexity of college-level mathematics problems, with the aim of applying our framework to the automatic generation of such problems controlled computational complexity.

Providing students with suitably complex practice problems is crucial for motivating them and facilitating deeper understanding. Manually constructing such problems consumes time, which mathematics teachers can otherwise use to mentor students.

Many software tools and services for automatic generation of mathematics problems are available on the web, but they provide only materials up to high school level. In addition, no standardized methods are provided to evaluate and control the computational complexity of the generated problems. Among the popular web sites and services, we list some examples. Wolfram Problem Generator$^{TM}$ [1] and Davitily Math Problem Generator$^{TM}$ [2] deal with mathematics problems for high school students. SuperKids Math Worksheet Creator$^{TM}$ [3] deals with arithmetic problems for children attending elementary schools. However, our framework is new as it deals with math at the college level and introduces suitable methods for evaluating computational complexity from the learners' perspective.

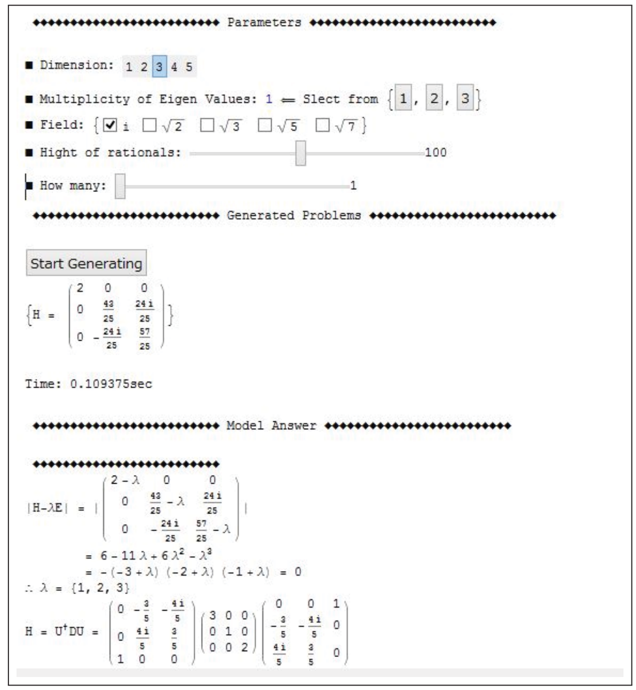



Fig. 1: GUI for generating eigenvalue problems

Figure 1 presents a graphical user interface (GUI) for generating eigenvalue problems using Mathematica$^{TM}$. Here, the user selects parameters such as the algebraic number field used in the calculation, the number of the calculation steps, and the matrix dimension , which determine the computational complexity of the generated problem. The user then selects the problem category and provides the required parameters that control the computational complexity. To avoid excessive selection for busy users, predefined sets of recommended parameters are also stored in the system, and thus ease selection for the user. Problems with the required complexity along with model answers are generated. Eigenvalue is a topic usually taught in linear algebra courses during engineering, however, most textbooks do not provide sufficient practice problems. As a result, teachers must create additional problems to be used in classes, assignment, and exams. In this paper, we present an automatic generation of diagonalization problems for Hermitian matrices to illustrate the relevance of our proposed framework.

## 2. Complexity from Learners' Perspective

Rigorous concepts of complexity are available in various forms in standard textbooks such as [5]. We deal with a different variety of complexity, subjective complexity, where complexity is measured by the difficulty that leaners' perspective. Designing practice problems that are sufficiently complex, but not excessively, are crucial for keeping a learner motivated. We propose a new framework for estimating such computational complexity and demonstrate its relevance by developing a framework for automatic generation of complexity-controlled practice problems. Our framework enables us to

1) control the number of the calculation steps,
2) limit the height of rational numbers involved in a calculation, and
3) deal with algebraic numbers.

The computational complexity of a generated problems is defined as the approximate sum of the heights of the rational numbers (the maximum ratio of the absolute values of the denominator and numerator) appearing in its model solution. In the hope of extending our work to other mathematics problems, we incorporated algebraic number fields in our system. The user can select the calculation field from the rational number field and other algebraic fields extended by irrational numbers, especially, quadratic irrational numbers, and fourth-power irrational numbers.

## 3. Automatic Generation of Eigenvalue Problems

Eigenvalue problems are usually taught in linear algebra courses during engineering. Eigenvalue problems appear in two forms: diagonalization of Hermitian matrices and Jordan canonicalization of linear transformations. In this paper, we deal with diagonalization of Hermitian matrices. The process of generating complexity-controlled eigenvalue problems includes

1) predefining unitary matrices,
2) generating eigenvalues,
3) generating $n$ Hermitian matrices, and
4) selecting matrices suitable for exercises.

We explain each step in detail. First, we generate nearly the entire set of tractable unitary matrices and classify them by algebraic number fields. We define an original function that can extract all of the irrational numbers appearing in the entries of a tentatively generated matrix. For example, this function returns the list $[\sqrt{-1}, \sqrt{2}, \sqrt{3}, \sqrt{5}]$, where

$$\begin{pmatrix} \sqrt{-5} & \sqrt{2} \\ \frac{1}{\sqrt{3}} & 7 \end{pmatrix} \qquad (1)$$

is entered as a primary material that is later subject to Gram-Schmidt orthonormalization. Diagonalizing a given Hermitian matrix requires

1) calculating eigenvalues,
2) determining the eigenvector that corresponds to each eigenvalue, and
3) constructing a unitary matrix using these eigenvectors.

Hence, the number of calculation steps does not vary once the dimension is determined. General Hermitian matrices can be generated from a diagonal matrix D and a unitary matrix U as

$$H = UDU^{\dagger} \qquad (2)$$

where $U^{\dagger}$ is the conjugate transpose matrix of $U$. Equation (2) is rewritten as $D = U^{\dagger}HU$. The most difficult part is generating a unitary matrix with the specified properties. However, the number of matrices suitable for this purpose is relatively small because the entries of those matrices must be obtained from a given algebraic number field, and the heights of the involved rationals must be restricted, furthermore, all of the column vectors must form an orthonormal system. Therefore it is possible to predefine almost entire sets of unitary matrices that can be used to generate Hermitian matrices that can be diagonalized with specified complexity.

### 3.1 Generation of Unitary Matrices

This section describes the generation of $3 \times 3$ unitary matrices through examples. The procedure for generating a matrix comprises four major steps:

1) generating a unit column vector,
2) verifying that all the entries belong to the given number field,
3) constructing an orthonormal basis from two other linearly independent column vectors using the Gram-Schmidt procedure, and
4) verifying again that all of the entries of those basis vectors belong to the given number field.

In step 1, we generate various unit vectors that are of the form in Equation (3).

$$\boldsymbol{e_1}^t = (\pm\frac{i}{j\sqrt{k}}, \pm\frac{l}{m\sqrt{n}}, \pm\sqrt{1-((\frac{i}{j\sqrt{k}})^2+(\frac{l}{m\sqrt{n}})^2)}), \qquad (3)$$

where $i, j, l$, and $m$ are rational integers, and $k$ and $n$ are $2, 3, 5, 7$, or 1. In step 2, all of the irrational numbers such as $\sqrt{2}, \sqrt{3}$, and $\sqrt{-1}$ are extracted from the vector and matrix. We select the vectors whose entries belong to the specified number field. In step 3, $e_1$ and two other vectors are orthogonalized. Only an additional two vectors are required to form a linearly independent triple together with $e_1$. Hence, for easy calculations, we can take them from sparse matrices, where only the positions of nonzero entries are important. Figure 2 indicates that the field cannot be retained after orthogonalization.

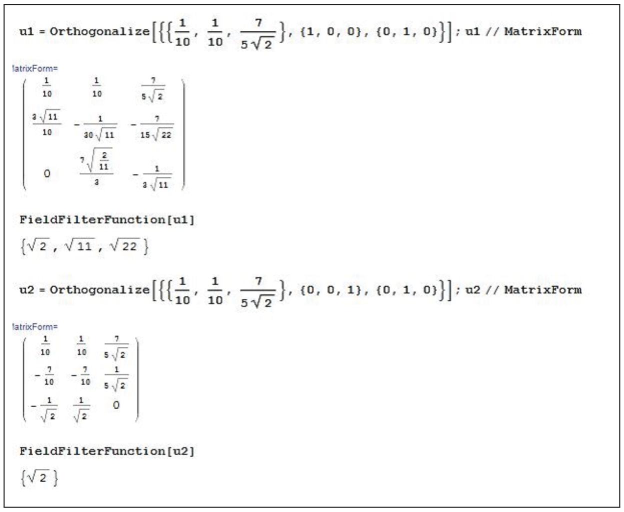



Fig. 2: Example of orthogonalization. The upper part shows failure in retaining the number field after orthogonalization in the case in which $(1,0,0)^t$ and $(0,0,1)^t$ are appended. The other shows success in retaining the field, when $(0,0,1)^t$ and $(0,0,1)^t$ are appended.

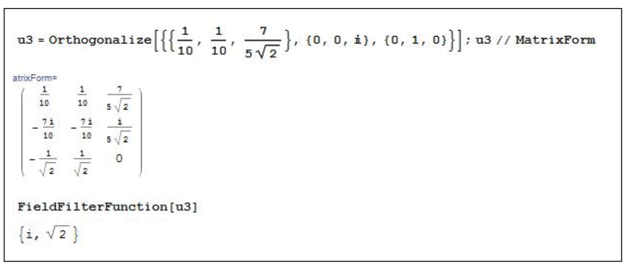



Fig. 3: Example of expansion of the orthogonalization to imaginary matrices.

We generated three patterns of unitary matrices by changing the position of the non-zero element of each vector. If the user wants a complex number field, then $\sqrt{-1}$ must be added in some entry of an initial vector (see Fig. 3). In step 4, the number field of the components is verified again. This step is necessary because Gram-Schmidt orthogonalization takes square roots which may cause further algebraic extension of fields. We select matrices all of whose entries have rationals of low heights in their subexpressions. These forms the basic set of tractable unitary matrices. Of the 500,000 generated unitary matrices in a preliminary stage, the filter selects 681 matrices according to the criterion described in later sections. The number of predefined matrices are listed in Tables 1 and 2. Though the basic set is relatively small (681), we can generate other tractable matrices by multiplying them among themselves and by taking direct sums as follows: given two matrices of the same dimension

$$U_1 \text{ and } U_2 \in U(n), \tag{4}$$

Table 1: Predefined orthogonal matrices

| Field \ Dimension | $2\times 2$ | $3\times 3$ | $4\times 4$ |
|---|---|---|---|
| $Q$ | 65 | 20 | 4245 |
| $Q(\sqrt{2})$ | 27 | 33 | 2503 |
| $Q(\sqrt{3})$ | 12 | 10 | 930 |
| $Q(\sqrt{5})$ | 20 | 25 | 1714 |
| $Q(\sqrt{7})$ | 12 | 3 | 927 |

Table 2: Predefined unitary matrices

| Field \ Dimension | $2\times 2$ | $3\times 3$ | $4\times 4$ |
|---|---|---|---|
| $Q(\sqrt{-1})$ | 129 | 20 | 25066 |
| $Q(\sqrt{2},\sqrt{-1})$ | 65 | 33 | 8488 |
| $Q(\sqrt{3},\sqrt{-1})$ | 30 | 11 | 2862 |
| $Q(\sqrt{5},\sqrt{-1})$ | 68 | 26 | 9068 |
| $Q(\sqrt{7},\sqrt{-1})$ | 24 | 5 | 2142 |

we obtain

$$U_1 U_2 \in U(n). \tag{5}$$

Given two matrices of possibly different dimensions

$$U_1 \in U(m) \text{ and } U_2 \in U(n), \tag{6}$$

we obtain

$$U_1 \bigoplus U_2 \in U(m+n). \tag{7}$$

Note that the number field involved is preserved under both multiplication and direct sum operations. For example,

$$\begin{pmatrix} -1 & 1 \\ 1 & 0 \end{pmatrix} \text{ and } \begin{pmatrix} \sqrt{-1} & 0 \\ 0 & 1 \end{pmatrix}. \tag{8}$$

Using such methods, we can obtain sufficient unitary matrices, as presented in Tables 1 and 2.

## 4. Demonstration

We demonstrate the automatic generation of eigenvalue problems for Hermitian matrices and evaluate the complexity of the generated problems. Figure 4 presents the results of ten generated Hermitian matrices. The dimension of each

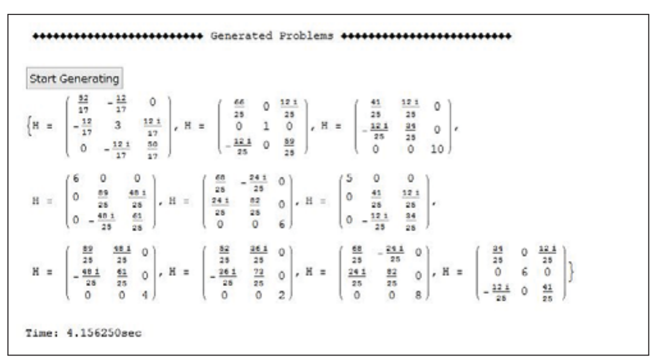



Fig. 4: Samples of generated Hermitian matrix

Table 3: Time (sec) for generating problem 1

| | $2 \times 2$ | $3 \times 3$ | $4 \times 4$ |
|---|---|---|---|
| $Q$ | 5.33 | 17.14 | 1615.72 |
| $Q(\sqrt{2})$ | 702.81 | 1185.03 | 768.2 |
| $Q(\sqrt{3})$ | 702.00 | 1324.84 | 810.72 |
| $Q(\sqrt{5})$ | 353.91 | 1677.19 | 484.39 |
| $Q(\sqrt{7})$ | 320.05 | 3154.67 | 1684.18 |

Table 4: Time (sec) for generating problem 2

| | $2 \times 2$ | $3 \times 3$ | $4 \times 4$ |
|---|---|---|---|
| $Q(\sqrt{-1})$ | 110.61 | 170.72 | 1712.11 |
| $Q(\sqrt{2}, \sqrt{-1})$ | 655.34 | 2175.56 | 809.42 |
| $Q(\sqrt{3}, \sqrt{-1})$ | 599.29 | 4585.61 | 924.39 |
| $Q(\sqrt{5}, \sqrt{-1})$ | 691.75 | 3439.33 | 500.33 |
| $Q(\sqrt{7}, \sqrt{-1})$ | 3543.17 | 22911.11 | 1821.26 |

matrix, multiplicity of eigenvalues, algebraic number field, height $h$ of numerical calculation, and number $n$ of problems is 3, 1, $\sqrt{-1}$, 100, and 10, respectively. The system generates $n$ problems on demand. Each matrix

1) has a maximum absolute value of involved rationals less than $h$,
2) belongs to specified number field, and
3) differs from already generated matrices.

In this case, generating ten problems took 4.16 seconds. The time measurements for generating problems are listed in Tables 3 and 4.

Tests were conducted generating 1,000 problems for each number field. Generating one problem takes 1.18 seconds on average. As the tables indicate, generation consumes more time when the number field is complex because is because the system needs to decompose all of the matrix entries to check the maximum height of the rational numbers.

As an experiment, we asked some college science students to solve the generated problems and some non-complexity-controlled (without our methods) problems. On average, the former case, students required 2 min to calculate eigenvalues and 3 min to construct a unitary matrix, while for the latter case students required 6 min to calculate eigenvalues and 7 min to form a unitary matrix. These results demonstrate the validity of our proposed framework. Large-scale verification experiments will be conducted out with the cooperation of university teachers.

## 5. Conclusion

In this paper, we developed a new framework for evaluating and controlling computational complexity from the learners' perspective. In addition, we developed an automatic generation system for eigenvalue problems according to our proposed framework. The automatic generation of complexity-controlled eigenvalue problems is one of the sample implementations that validate our proposed framework. Controlling the complexity of eigenvalue problems involves restricting the number of calculation steps, the height of the involved rationals, and the algebraic number field appearing in a model solution. Small-scale experiments illustrated the relevance of our proposed framework. Constructing problems with sufficient computational complexity is essential for maintaining learners' motivation. Our framework helps teachers prepare learning materials and thereby save time for mentoring students.

We expect our technique to be applicable to other subjects in linear algebra and analysis. Our system paves a path to strict quantitative control on various aspects of complexity from the learners' perspective. Future work will include the application of our methods to other areas such as differential equations, and number theory. We also intended to show that this system has a positive effect on college-level engineering education. Therefore, demonstration experiments in classroom will be the next step of our research.

## References

[1] Wolfram Research (2012), Wolfram Problem Generator, <http://www.wolframalpha.com/problem-generator/>, (Access: 2013/4/17).

[2] David Pardue and Ly Nguyen (2007), Davitily Math Problem Generator, <http://www.mathproblemgenerator.com/>, (Access: 2013/4/17).

[3] Andrew Maisel (1996), SuperKids Math Worksheet Creator, <http://www.superkids.com/aweb/tools/math/>, (Access: 2013/4/17).

[4] LLC (2000), Math Fact Cafe, <http://www.mathfactcafe.com/worksheet/buildit/>, (Access: 2013/4/17).

[5] Michael Sipser (1997), Introduction to the Theory of Computation, PWS PUBLISHING COMPANY.

[6] Wolfram Research (2009), Wolfram Mathematica 9 Documentation, <http://reference.wolfram.com/language/>, (Access: 2013/4/20).

[7] Geza Schay (2012), A Concise Introduction to Linear Algebra, Brikhauser; 2012 edition.

[8] Henry Ricardo (2009), A Modern Introduction to Linear Algebra, Chapman and Hall/CRC

[9] Paul J.McCarthy (1996), Algebraic Extensions of Fields, Dover Publications, Inc., New York.

[10] James.H.Wilkinson, (1988), The Algebraic Eigenvalue Problem (Monograph on Numerical Analysis), Oxford University Press; 1 edition

[11] John William and Scott Cassels (2010), Algebraic Number Theory, London Mathematical Society

[12] Sandy Kemsley (2000), abc teach, <http://www.abcteach.com/>, (Access: 2014/6/21).

[13] Ben Taylor Langton (1999), Quick Math, <http://www.quickmath.com/>, (Access: 2014/6/21)

# Cognitive Prior-Knowledge Testing Method for Core Development of Higher Education of Computing in Academia

**Mohit Satoskar**[1]
[1]Research Associate, Department of Computer Science and Information Technology,
Ramrao Adik Institute of Technology , Navi Mumbai, Maharashtra, India

**Abstract** - *Aspects and necessity of testing students' real prior knowledge before admission to higher education is analyzed. Prior-Knowledge testing method presented along with inter-dependency of students' knowledge testing components with time machine. Efficiency of students prior knowledge testing is described. It is clearly analyzed how entrance exam techniques affect education system to a larger extent. Algorithm of Cognitive prior-knowledge testing method is briefly highlighted. Three knowledge testing components(Objective, subjective and practical model) are described with hierarchy of structural model. Virtual Lab technique of Information and communication technology(ICT) is presented. The percentile technique is presented to analyze the result for the admission process. It is shown how this method will improve the education system and reduce unemployment rate.*

**Keywords:** prior-knowledge testing method, virtual lab, computing percentile, information and communication technology.

## 1 Introduction

At present, technological innovation in higher education, particularly the development of educational standards have become core components of development. The general stance was that, significant new technological development proceeded gradually, as, higher educational institutes could not foretell, that the impact of computing education would be needed, at a core of the change. In the battle of quality, many institutes are taking part in developing intensive methodological work. At the same time, the study of computer science can help to design framework, making possible, fully testing of prior-knowledge base with efficiency. These steps will help the higher education system change from mediocre to world-class. It is interesting to note that, this method will provide cream of excellent students who have the right to deserve the admission for higher education. "*Education is not the filling of a vessel but the kindling of a flame"*- Socrates

It is necessary to work-out prior-knowledge testing method to produce fruitful results to the industry and academia.

Generic root cause:

- student assessment without structural traditional method of entrance exam.
- It will override the so called traditional assumption method of prior knowledge of student.

In many cases, admission to higher education can be given by analysing, judging, testing the core concept, prior to admitting the student[5][6]. In result, it will strengthen the level of education, its quality and automatically it will reduce unemployment to some extent.

In fact, entrance exam technique is not scalable as it also carries small scale syllabus. Instead of that, prior-knowledge testing method can give productive result.

## 2 Prior Knowledge Testing Method for Academia

Many higher educational institutes boast of their robust education system which has given birth to exemplary entrepreneurs. Continuing with its trend of multiple educational institutions it retains its core speciality i.e. having innumerable types of courses and a customised fee-structure in the educational institutes. Major higher educational institutes offering education to the students based on certificates/ grades don't prove that students are capable for the particular course. It is a traditional method which is based on assumption and presumption technique.

While offering higher education, it is need to test the student's prior-knowledge base of the specific area. As this problem arose, system got covered by the most typical aspect of 'entrance exams'. However, entrance exam is based on set of syllabus and students studies that set for temporary.

Interestingly, it is seen that, even after graduating, students fail to succeed[5][6]. Importantly, prior-knowledge testing should be clearer and the real knowledge should be tested. However, very limited numbers of students are getting fruitful results upon completion of course.

## 2.1 Analysis and calculation of functional collaborative node

In every walk of life there are certain components present either solely or in the necessary combination namely science, technology, engineering and mathematics. Science can be defined as knowledge about any particular parameter based upon the facts and functions learnt through experiments and the inferences drawn there from. Whereas technology can be stated as the means of application of the acquired knowledge in that particular area. Engineering is the most important function of all. It directs all the personnel and guides other functions to work in a direction to achieve some fruitful results. And mathematics implies the logical functionality. These four functions are omnipresent either in a singular form or in combination [2][3][11].

All these individual functions are of utmost importance but for the optimum results all these four functions should work hand in hand. Proposed model is comprising of all these functions which will help us achieve enormous success. STEM model is a medium to test the logical, theoretical as well as the practical knowledge of the pupil along with its optimum utilization[3]. It is a technology based model which will be platform independent and user friendly. This model will test the theoretical as well as the practical knowledge of the student in the corresponding areas.

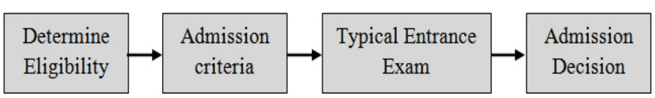



*Figure 1 A typical model of admission process*

The model shown in Figure 1 suffers from poor analyzing and testing the logic of students prior-knowledge acquisition. Characteristics such as admission criteria, for example 60% is mandatory in previous education for further education does not prove that students have real knowledge. Entrance exam also is not entirely convincing since it ignores certain crucial parameters such as syllabus. Student may study limited amount of given syllabus of entrance exam which however doesn't prove whether they have knowledge or not? Thus it is required to highlight guideline of syllabus for better execution. In addition to that specific percentage criteria lacking to test prior base knowledge. More logical might be the sequence presented in Figure 2.

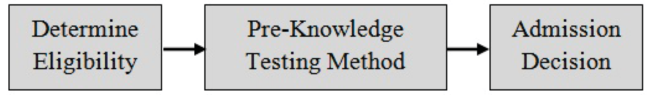



*Figure 2 Modified model of admission process with prior knowledge testing method*

It is shown that there exist three channels of knowledge delivery: via text, via audio, via video [2]. To test these delivery channels of knowledge, rigorous formation of testing components will strengthen the admission process of higher education.

**Strategic Maths**

At present, entrance examinations are the part and parcel of current education system. It often also proves to be a barrier to the real potential of a pupil. In certain circumstances, student is unable to act accordingly leading him to the ultimate failure in such examinations. Though a pupil fails to pass the examination with flying colors, does not necessarily implicate that there is some weakness in his knowledge acquisition.

Thus the efficiency of the knowledge has to be tested for the realization of the actual potential of that pupil. Efficiency is the function of application of the acquired knowledge with a close relation to the variable of the time taken for such application.

$$time\ taken(tt) = time\ alloted(ta) - time\ saved(ts) \quad (1)$$

$$efficiency = \frac{question\ covered + accuracy}{time\ taken} \quad (2)$$

In the proposed model for every question stated maximum time that has to be given for the solution of that corresponding question will be stated. Such time can be named as time allowed. Time saved will be the difference between the parameters of time allowed and time taken as we have elaborated in equation 1. Such time saved will determine the efficiency of such student. We can say that the time saved function is directly proportional to the function efficiency.

To this dependant function to work there is a variable playing a vital role named accuracy. Time saved along with the accuracy of the solution will ultimately determine the efficiency of the student. The relativity of the time taken and the accuracy of such solutions with the efficiency of the learner is shown in the equation 2.

In the above stated equation the questions covered are the number of questions which are answered correctly. The variable of accuracy is determined by the certain parameters such as use of certain keywords, also the order of preferential importance given to such keywords. Thus the accuracy also will be measured cardinally. And the sum total will be thus divided by the constraint of time taken. The greater will be the denominator the lesser will be the outcome which is the efficiency and vice versa.

## 2.2 Formal structure of the prior-knowledge testing method

In the proposed model there are various components already defined. For the testing of prior-knowledge, all the necessary aspects should be tested accordingly. Current traditional entrance examinations only test the theoretical knowledge possessed by the student. This assessment model fails in the real goals of assessing the knowledge that has been acquired as well as understood and not just acquired knowledge. This constraint is based upon the vital concept of efficient practical application of the acquired knowledge.

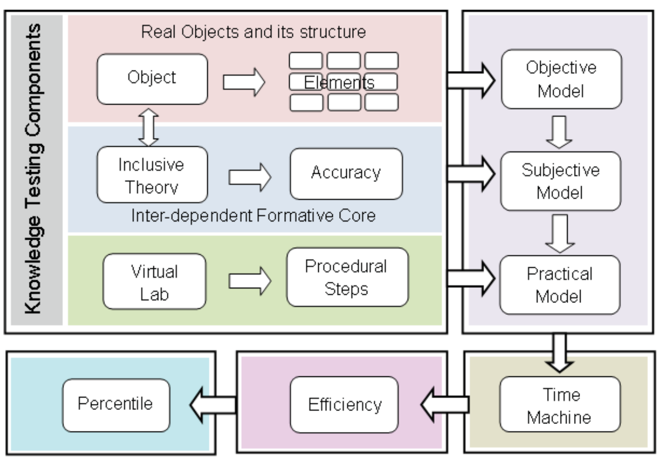



*Figure 3 Structural representation of prior-knowledge testing method*

In the above prior cognitive knowledge assessment method, the crucial difference is that this model aims at judging the practical knowledge of the learner. In this model question bank will be based within a particular framework which will be guided by the defined syllabus. Questions will be of all the three types namely objective subjective and practical. Objective questions will attempt to test the logical understanding of the learner along with the immediate application of such logical reasoning. Whereas the theoretical also named as the subjective type of questions will judge the actual understanding of the learner of that particular area of knowledge. The whole model will be '*Online Computer Based*'.

In this type the constraint of accuracy is also as much important as the constraint of time taken is. The last will be the practical testing where the question will be presented virtually to the learner for solution. The practical model will be based on Virtual lab technique. Virtual lab will provide to the students, the result of the experiment of modeling the physical phenomenon and carrying out simulations to generate the result of a particular real-world experiment [12]. In this kind the practical application of the knowledge along with the conceptual clarity will be tested vigorously.

All these three types of questions will be assigned a general time frame in the model. Taking such time into consideration, time taken and time saved will be decided. And furthermore combining the celerity with the accuracy of such celerity will conclude the efficiency and ultimately the eligibility of the learner for entering to the corresponding institution in corresponding course.

## 2.3 Algorithm of cognitive prior-knowledge testing

To test whether student have required pre-requisite knowledge or not, implementation of following algorithm is necessary. Model is classified in three components in which objective and practical model will be tested instantly by computer. Inclusive theoretical writing model will be redirected to expert for rigorous testing.

1) initially select objective model component weighing of 25% interdependent on time machine of 30 minutes in section adaptive test.
2) select subjective model component weighing of 25% separately interlinking with time machine of 30 minutes.
3) submit and redirect subjective component to individual experts for testing.
4) select last component of section adaptive test - virtual lab based practical model, weighting 50% interlinked with time machine of 60 minutes
5) test objective model and practical model of computer based test on the spot and generate result using percentile formula.
6) For each expert, test subjective writing model on the basis of language proficiency, accuracy, core content and structure and calculate the result on the scale of language modelling bands out of 5.
7) generate result in percentile and select prospective students as per their ranks for higher education studies.

# 3 Technique of Computing Percentile

In the proposed model of assessment the scores are to be assigned according to the percentile pattern and not the ranking method. percentiles can be defined as: *p*th percentile of the distribution, states, the percentage of the distribution, that is, less than or equal to pth value. Following stated formula will elaborate the working of the calculation of percentiles pattern.

$$percentile = \frac{number\ of\ values\ below\ x}{n} \times 100 \qquad (3)$$

Here the scores of every learner will be computed and will be entered into the database and they are arranged according to the proposed order that is the ascending order. Then for a particular percentile for the corresponding is calculated by dividing the number of values lesser than that particular score by the total number of the students. And the outcome then has to multiplied by 100. Thus the percentile will show the ordinal position of that student.

# 4 Dependency of prior-knowledge with unemployment rate

Dr A.P.J Abdul Kalam admits: *"It's not about how many engineers we produce, but how many engineers we produce with employability"*. Most engineering graduates are indeed unemployable in various regions of the world. India churns out tens of thousands of graduates each year but less than half of them are “employable” or possess the basic skills necessary for any industrial role, says a report [5]. Youth unemployment and underemployment is prevalent around the world because young people lack skills, work experience, job search abilities and the financial resources to find employment (United Nations, 2003; ILO, 2006; Matsumoto et al., 2012).

The variable of unemployment is directly proportional to the variable of quality of prior-knowledge which may impose the significant cost of national budget [6]. A landmark report on youth unemployment in the UK claims that the ‘net present value of the cost to the Treasury, even looking only a decade ahead, is approximately 28 billion (GBP)’ (The ACEVO Commission on Youth Unemployment, 2012:4). To impose quality education and better employment, prior-knowledge testing is needed.

# 5 Conclusions

Improving education system is a serious concern. Ironically, as increase in educational institute will fulfil the dream of 'serving generic education' , institutes are making money rather than distributing quality improved education. Knowledge will be tested in this model. Education shouldn't be treated as business, there will be massive a imbalance. Automatically it will sustain employability.

New solutions to the problems of real time knowledge testing on the battlefront were designed. Such major changes need to be adopted by myriad higher educational institutes for a comprehensive superlative education admission process. It is a need to adopt this model in the higher education system, both access, completion, and excellence, are getting the attention they need to test cognitive ability.

The three basic components of cognitive prior-knowledge testing component model is presented. With the help of virtual lab technique real world practical knowledge of students can be tested more efficiently. This model can be implemented with the help of ICT which will also try to minimize the underemployment of the computer graduates.

# 6 Acknowledgements

This work was supported by Computing Research Lab, Ramrao Adik Institute of Technology, Navi Mumbai - India. The author gratefully acknowledges the support from Miss. Anuprita Nitin Satoskar during undertaken study.

# 7 References

[1] Charnine, M., and V. Protasov. "Optimal Automated Method for Collaborative Development of University Curricula." *ICAI'13- The 2013 International Conference on Artificial Intelligence*, Las Vegas, U.S.A. pp 96-100. Vol. 1

[2] Satoskar, M. "An Enhanced Evaluation Approach: Computer Science and its Framework." *Proceedings of the International MultiConference of Engineers and Computer Scientists*. Vol. 1. 2014., Hong Kong. ISBN: 987-988-19252-5-1.

[3] Satoskar[1], Mohit, and Sagar Mali. "Comprehensive Curriculum of Programming for Engineers, its Teaching Models and in-Lab Monitoring Technique."*Proceedings of the International MultiConference of Engineers and Computer Scientists*. Vol. 1. 2015. ISBN: 978-988-19253-2-9

[4] Rubina Lamba, "A Study of Examination System of AICTE", International Research Journal, Vol I, Issue 10. September 2010. ISSN- 0975-3486.

[5] Aspiring Minds - Report on Unemployment in India. June 2013. New Delhi.

[6] Matsumoto, Makiko, Martina Hengge, and Iyanatul Islam. "Tackling the youth employment crisis: A macroeconomic perspective." (2012). International Labour Organisation. 2012.

[7] "National Focus Group on Examination Reforms", National Council of Educational Research and Training, 2006. ISBN: 81-7450-545-8.

[8] Director, Stephen W., et al. "Reengineering the curriculum: design and analysis of a new undergraduate electrical and computer engineering degree at Carnegie Mellon University." *Proceedings of the IEEE* 83.9 (1995): 1246-1269.

[9] Diamond, Robert M. *Designing and Improving Courses and Curricula in Higher Education: A Systematic Approach*. Jossey-Bass Inc., Publishers, 350 Sansome St., San Francisco, CA 94104-1310, 1989.

[10] Qiao-fang, Zhao, and Li Yong-fei. "Research and Development of Online Examination System." *Proceedings of the 2012 2nd International Conference on Computer and Information Application (ICCIA 2012)*. France

[11] Cheney, Gretchen Rhines, Betsy Brown Ruzzi, and Karthik Muralidharan. "A profile of the Indian education system." *Prepared for the New Commission on the Skills of the American Workforce* (2005).

[12] Manual of Engineering Labs (Virtual Labs), *Ministry of Human Resource Development, Government of Indi*a *under National Mission of Education.* Retrieved from: https://www.youtube.com/watch?v=mVBYNXLaDdg (accessed: 21 March 2015)

# Visualization of Learning Situation of C Language Using Self-Organizing Maps

**Tomohiro Enda**[1], **Hiroshi Dozono**[1], **Masanori Nakakuni**[2]
[1] Faculty of Science and Engineering, Saga University, 1-Honjyo Saga, 840-8502 JAPAN
[2] Information Technology Center, Fukuoka University, 8-19-1, Nanakuma, Jonan-ku, Fukuoka 814-0180 JAPAN

**Abstract**— *The lecturer is difficult to identify the student's behavior and learning situation in programming lecture. In this research, acquiring behavior information of students from the keyboard and mouse, and the visualization is performed by clustering using SOM. The proposed system can identify the learning situation of the students. In the future, the development of systems that can be used in real time during the lecture is expected.*

**Keywords:** Education of programming, Computer educational support, Self Organizing Map(SOM)

## 1. Introduction

In the lecture which uses computers as programming lecture, it is difficult for the lecturer to identify the learning situations of the students from the behaviors of them and computer screens of themselves. The software which can monitor the students' screens as array of small screens can be available as shown in Figure 1[1]. however it is difficult to

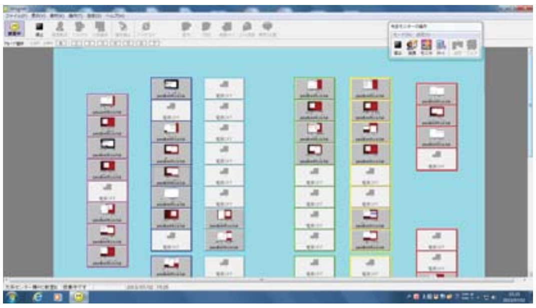


Figure 1: Wingnet

monitor each student. Such software can display the screen of each student, however it is difficult to monitor all students at once. We reported the experiments of identifying the learning situation using keyboard and mouse inputs obtained from the computer of each student. We are developing the practical application which can be used in the lecture for identifying the learning situation more effectively. In this paper, the improvements for identifying the relations among the students is proposed, and the experimental results are shown.

## 2. Self Organizing Maps

Self-Organizing Map (SOM)[2] is the model of the neurologic function of the cerebral cortex developed by T.Kohonen. As the class of neural networks, SOM is the feedforward type of two hierarchical without the interlayer using algorithms of unsupervised learning.

SOM converts a nonlinear and statistics relations that exist between higher dimension data into the image with a simple geometrical relation. They can usually be used to make a higher dimension space visible because it is displayed as the lattice of the neuron of two dimensions. Moreover, it becomes easy to be able to visualize higher dimension information because it is possible to cluster without the preliminary knowledge, and to understand the relations among these multi-dimensional data intuitively for human. SOM is

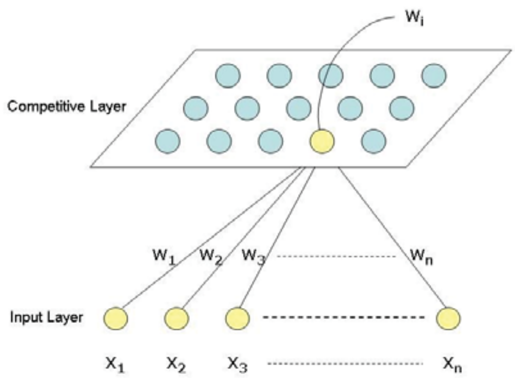



Figure 2: Self Organizing Map

basically a network in the double-layered structure in the input layer and the competitive layer as shown in Figure 2. However, there are not connections between neurons in the same layer. The first layer is input layer $\mathbf{x}$ of n dimension, and the second layer is called a competitive layer, and is generally two dimension array for visualizing output data. It is assumed that it has the neuron that input data is given by n-dimensional real number vector $\mathbf{x} = (x_1, x_2, ..., x_n)$ and two dimensions SOM were arranged on the grid point of $m * m(= M)$piece. Input data $\mathbf{x}$ is presented to all neurons. The neuron located in $(j, i)$in two dimension lattice array has $\mathbf{w_{j,i}} = (w^1_{j,i}, w^2_{j,i}, ..., w^n_{j,i})$ weight vectors which are tunable corresponding to the input vectors. This $w_{j,i}$is called as reference vector. The algorithm can be divided into two phases in a competitive phase and the cooperative phase. First of all, the neuron of the closest ,in a word, the winner neuron, is selected at a competitive phase. Next, the winner's weight are adjusted at the cooperation phase as well as the lattice near the winner. Figure 4 shows the animal map which is organized by using the animal data shown in Figure 3 as input vector. Each animal is represented by a vector which denotes the feature of the animal such as size, number of foot, swim or not swim and etc. The map size is $10 \times 10$. The animals are arranged on the map according to

| | Ele1 | Ele2 | Ele3 | Ele4 | Ele5 | Ele6 | Ele7 | Ele8 | Ele9 | Ele10 | Ele11 | Ele12 | Ele13 | Ele14 | Ele15 | Ele16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | Small | Middle | Big | Noctur | 2fot | 4fot | Hair | Ungulat | Mane | Shuttle | Stripes | Hunting | Run | Fly | Swim | Herbivo |
| Dove | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0.5 |
| Fox | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0.5 |
| Hen | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0.3 | 0 | 0 | 1 | 1 | 0.5 |
| Lion | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0.5 |
| Goose | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| Eagle | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| Dog | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| Wolf | 0 | 1 | 0 | 0.5 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Zebra | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| Duck | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Cat | 1 | 0 | 0 | 0.5 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Owl | 1 | 0 | 1 | 0.5 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 |
| Tiger | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Horse | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 |
| Hawk | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| Cow | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Figure 3: Animal data table

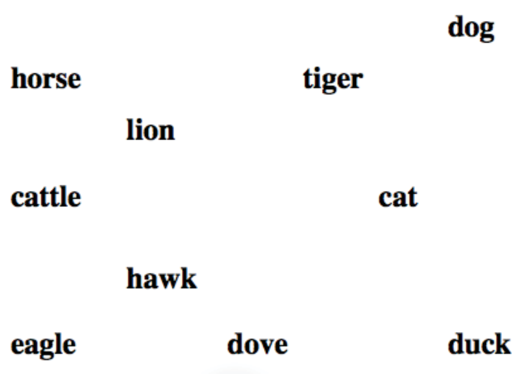



Figure 4: Animal map

their similarities of multi-dimensional input vectors which represents the feature of the animal.

We have applied SOM to the analysis and authentication system of behavior biometrics and analysis of the DNA sequences[3][4].

# 3. Acquisition of student behavior

## 3.1 Method for acquiring student behavior

The program coded in HSP(Hot Soup Processor)[5] is used to acquire the behavior data. HSP can scan the key-board, mouse and other devices as key code using getkey function. Table 1 shows the key codes assigned for mouse and keyboard.

Table 1: key codes table

| key code | pressed key | | key code | pressed key |
|---|---|---|---|---|
| 1,2 | left and right mouse botton | | 48-57 | 0-9 |
| 8 | back space | | 65-90 | A-Z |
| 13 | enter | | 96-105 | ten key |
| 46 | del | | 112-121 | F1-F10 |

## 3.2 Acquired data

The acquired key codes are sent to host computer, which collects the behaviors of all students, in every 5 seconds. The format of the data is shown as follows.

```
20140527171900,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
54,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,9,0,0,0,0,0,0,0,
0,0,0,0,0,12,0,0,0,0,0,13,11,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

The first element denotes the time when the data is acquired from year to second. In this case, this data is acquired at 2014/5/27 17:19:00. From 2nd to 121th elements, each element correspond to the key code. In this case, 73th element 8 denotes that the key "I" is pressed 0.08[s] in this samplel of 5[s]. For all keys, the accumulated time of pressing the key is recorded.

## 3.3 Input vector of SOM

The input vector of SOM is composed as follows.

1) Remove the 1st element time information.
2) Convert the element to binary value with replacing the value except 0 to 1, and leaving 0.
3) For each 6 data which represents the behavior of 30[s], calculate logical OR.

Time information is meaningless for this analysis, thus it is removed. The period of pressing keys may be available to identify each student, however in this research, the data of presses key is required, thus it is converted to binary value. In this research, the interval of analysis is taken as 30[s] considering the time to type keyword of C language for Japanese students. The converted input vector for SOM is 121 dimensional vector as follows.

```
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
0,0,0,0,0,1,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

# 4. Experiental result (1)

## 4.1 Setting of the system

As mentioned before, the behaviors of the students are collected to the host by the program written in HSP, and the analysis using SOM is conducted by the program developed on Visual studio 2012 using C language. For displaying maps, HTML is used considering the future extension of the system.

## 4.2 Content of the lecture

In this lecture, the students were given the problem to code the following program.

```
#include <srdio.h>
int main(void) {
int m;
printf("input integer number:");
scanf("%d",&m);
if(m%2) puts("Odd number");
else puts("Even number");
return 0;
}
```

This lecture is very early step of C programming, including input-output functions printf, scant and puts, and conditional branch using if-else structure.

## 4.3 Results of analysis using SOM

At first, the analysis of the students' behavior using the input vector for every 30[s] as mentioned in this section is conducted. Figure 5 shows the result for the just beginning of lecture. In this figure, each number denotes student's number

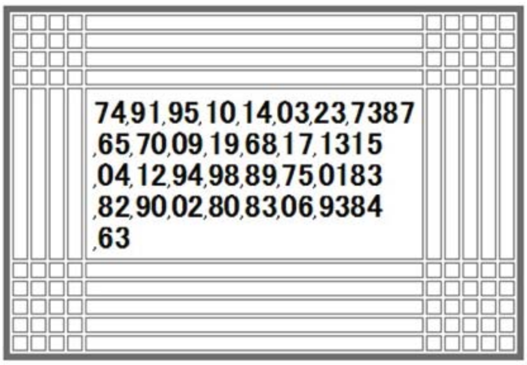



Figure 5: Map for beginning of the lecture

which is given to each student. All students are clustered on a unit because they did not using computers. Figure 6 shows the result for beginning of the programming of the given problem. The color of the number represents the corresponding input of the student shown in Table 2. The string is identified from the acquired data during 30[s]. The numbers spread on the map, and many numbers are colored

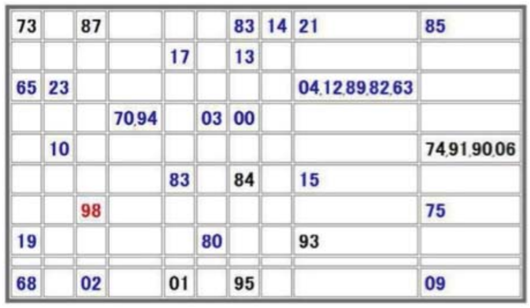



Figure 6: Map for beginning of the programming(1)

Table 2: Color table

| action | input |
|---|---|
| mouse click | blue |
| string "main" | red |
| string "printf" | yellow |
| string "return" | magenta |
| string "scant" | grey |

in blue, because the students began to program individually with clicking mouse. The student numbered as 98 had already inputted "main". On the other hand, some students represented as black numbers did not start programming. Figure 7 shows the result for middle of the programming. The students in the circle drawn on upper right of the

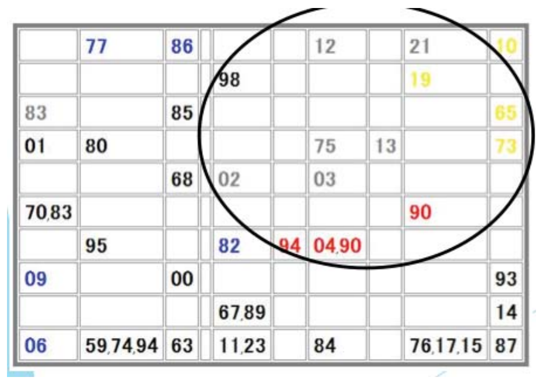



Figure 7: Map for middle of the programming(1)

map are colored in red, yellow and gray, which represent "main", "printf", and "scanf", and they are smoothly coding the program. In this result, the other students can not be identified the situation because this map is organized using the temporal information is 30[s]. Figure 8 shows the result for ending of programing. The most of the students clustered in a units, because they stopped to use computer. Some students who are using computers are clustered in other units. However, it is not clear that which students could program the problem, because the temporal information is used in this analysis.

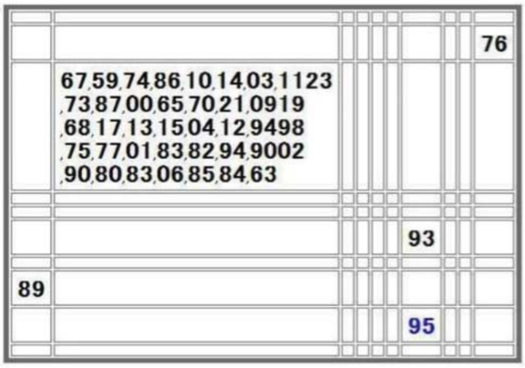



Figure 8: Map for ending of the programming(1)

## 5. Improvement of the analysis method

As shown in the previous section, the temporal behavior of the students can be clustered on the map, and rough estimation of the situation of the students can be haven. However, the history of the student behavior is not considered. For this problem, 3 improvements are conducted. The first improvement is added to the method for composing the input vector of 30[s] from the acquired data of 5[s]. In the

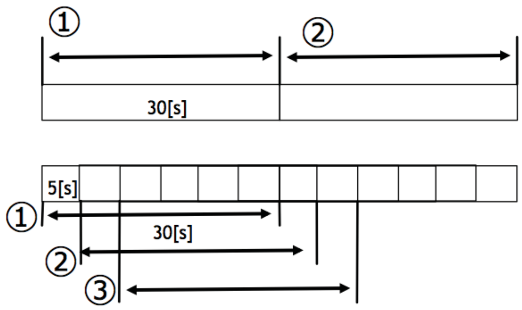



Figure 9: Composition of the overlapping input vector

previous experiment, the input vectors were composed as shown the upper side of Figure 9, and it is improved as shown in lower side with overlapping and sliding in every 5[s].

2nd improvement is the extension of the input vector. To keep the history of the student, the input vector is extended as to include the counts of the inputted C language keyword. In this experiments, the 121 dimensional input vector is extended to 125 dimension, with adding 122-125th elements representing the counts of "main", "printf", "return", and "scanf".

3rd improvement is taking the record of the relationship among students. The students, who are mapped closely by SOM, made similar behaviors in the period. In this experiment, for each student, the student who is the most closely mapped to the student is counted in the incidence matrix in each iteration. And, for each student, the students who has the strongest relationship with the largest value in incidence matrix is displayed as suffix on the map. With these 3 improvements, the history of the student behavior is considered to be visualized on the map.

## 6. Experimental result (2)

The experiments are conducted again using improved method for the analysis. Figure 10 shows the result for beginning of the lecture. As same as the result of the

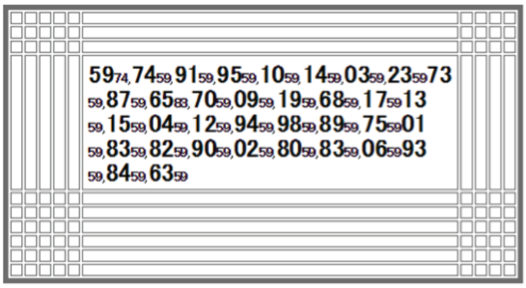


Figure 10: Map for beginning of the lecture(2)

previous experiment shown in Figure 5, the students are clustered in a unit, and they have relationship with the student in the same unit as shown by the suffix number. Figure 11 shows the result for middle of the programing. The students colored in yellow and colored in red inputted

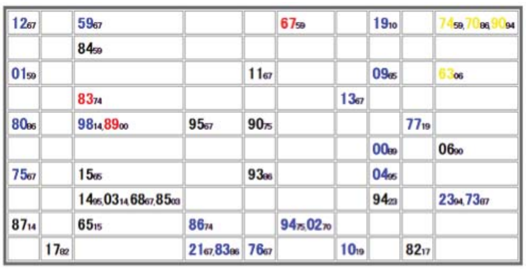


Figure 11: Map for middle of the programming(2)

"printf" and "main" at this period, and the students who have relationship with them shown by suffix are considered to be smoothly coding the program. Figure 12 shows the result for ending of the programing. Almost of the students are clustered in a unit and have relationship among them. On the other hand, the students on the upper left side does not have relationship with the clustered students. They are considered to fail to code the program, or to be lazy away during the lecture.

## 7. Conclusion

In this paper, the method foe visualizing the student behaviors and relationship among them using Self Orga-

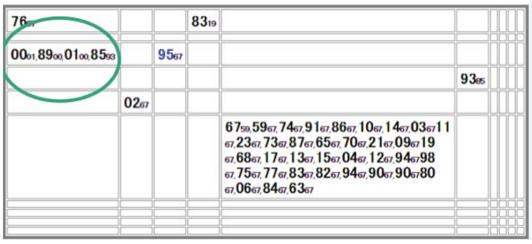


Figure 12: Map for ending of the programming(2)

nizingMap(SOM) is introduced. And three improvements to visualize the history of the behaviors are proposed. The experiment of visualizing the behaviors of the students acquired by the program written in HSP is conducted, and the situation of the students and their relationship can be identified from the organized map.

As the future work, the visualization of the relationship should be improved because that using suffix number is not easy to be understood. And the real time visualization system which can be used in the lecture, and which can help the lecturer and students should be developed.

# References


[1] (c)Computer Wing: Wingnet : Support for education, http://www.cwg.co.jp/

[2] T. Kohonen:Self Organizing Maps, Springer, ISBN 3-540-67921-9, 2001

[3] H. Dozono, M. Nakakuni: An Integration Method of Multi-Modal Biometrics Using Supervised Pareto Learning Self Organizing Maps, Proceedings of 2008 International Joint Conference on Neural Networks, pp,603-607, IEEE Press, Hongkong(2008)

[4] H. Dozono:An Algorithm of SOM using Simulated Annealing in the Batch Update Phase for Sequence Analysis, Proceedings of 5th Workshop on Self Organizing Maps 2005, pp. 171–178

[5] ONION Software: Hot Soup Processor Page, http://hsp.tv

# SESSION

# LATE BREAKING PAPERS: COMPUTER SCIENCE AND COMPUTER ENGINEERING EDUCATION

## Chair(s)

### TBA

# Software architectures for teaching induction heating principles

Alistair A. McEwan
Department of Engineering
University of Leicester
Leicester LE1 7RH, UK
Email: alistair.mcewan@le.ac.uk

Glen Mason
Faculty of Technology,
University Centre Peterborough,
Peterborough, PE1 4DZ
Email: glen.mason@peterborough.ac.uk

***Abstract*—Electromagnetism is a key topic in the field of electrical engineering and whilst induction heating is a common application of electromagnetism it is one that is often overlooked on many undergraduate degree programs. There are numerous reasons for this including the fact that key concepts of induction heating can be challenging for students to grasp and for lecturers to demonstrate due to two inherent problems. The first problem is the lack of an obvious visualisation method thus preventing a compositional approach to teaching. The second is that commercial induction heaters (which could be used to ameliorate the first issue) are generally costly, bulky, and require high power sources thus rendering them unsuitable in education environments. In this paper we present the development of an induction heater demonstrator that is being used in undergraduate teaching that attempts to address these issues. The key contribution of our demonstrator is in the architecture of the software used to control the heater—by presenting it as a control problem decomposed into a co-operative task set we are able to present the core concepts individually, and therefore achieve a compositional approach to presenting, and learning, the topic.**

## I. Introduction

The principles of electromagnetism are to be found in a wide variety of everyday applications. However despite this widespread appearance and use in engineering solutions, it remains a very difficult concept for undergraduate students to grasp. There are many common and well known areas in which the principles of electromagnetism are employed and exploited—for instance, electric motors, generators, relays, and so forth. However when asked to identify other—less immediately obvious—applications in which electromagnetism is exploited, students are often surprised to be introduced to examples such as loudspeakers, swipe cards, magnetic resonance imaging (MRI), and high speed trains.

In motivating students to study electromagnetism and appreciate it's pervasive nature in the real world, we have previously employed some well-known examples drawn from a multitude of undergraduate teaching texts(see, for instance [1], [2], [3]). Students find it particularly intriguing to realise that it would be impossible to see the stars at night were it not for electromagnetic fields, and that the power of electromagnetism is an area of active research as a future method of travelling through space[4]. Intrigue is further fuelled when it is explained that electromagnetic forces are $10^{36}$ times stronger than the Earth's gravitational field—and that if this were not the case all material would collapse into a black hole. The concepts that multiple forces exist, that these forces compete, and that (on Earth) electromagnetism wins this competition is one that interests students and that they find easy to conceptualise.

Despite the ease with which undergraduate students enthusiasm for the subject can be ignited, promoting a deeper understanding of the core scientific concepts in a structured and incremental manner remains a challenge. In our experiences of teaching to Engineering undergraduate students at University Centre Peterborough, one of the main challenges faced by students is in visualising, and conceptually separating, core concepts. These concepts are not usefully suited to typical transmissive approaches to learning and teaching (see [5] for related work)and are instead much more suited to a student-centric constructivist approach (such as presented in [6]). This is because it is not the understanding of the principles themselves that is the end goal in the learning and teaching process, but instead the ability to apply these principles to create novel and effective engineering solutions to problems.

Our motivation for the work presented in this paper is a consideration of a teaching and learning aid that empowers students not only to study the core principles of electromagnetism and more specifically electromagnetic induction heating, but to consider their application in the construction of an electromagnetic heating element in a way that supports the favoured student-centric constructivist approach. The contribution of the paper is the presentation of a novel design and architecture for an electromagnetic heater with associated control software that supports this. The primary novelty of our approach is in the design and architecture of the control software—it is designed around a time-triggered co-operative scheduling paradigm that enables us to separate out key phases on control, thereby enabling students to visualise and study each phase individually. To our knowledge this approach of separating concerns in the control software as a teaching aid is a novel contribution to the pedagogic challenges of teaching electromagnetism and electromagnetic induction heating and differs to works such as [7] in that our design focuses on pedagogy rather than simulating industrial equipment.

The paper is structured as follows: in Section II we present a brief resume of electromagnetism, the principles of electromagnetism, and electromagnetic induction heaters for the interested reader. We follow this in Section III with an introduction to the principles of time-triggered co-operative scheduling, and a descripition of our time-triggered task set and implementation in Section IV. We ruminate on the pedagogic aspects and benefits of this architecture relative to our original considerations and challenges in Section V. Finally, we draw some conclusions from our project in Section VI whilst also identifying some aspects for future work.

## II. A brief history of electromagnetism

Literature reports the history of understanding electromagnetism began in 1820, when Ørsted observed a compass needle moving when positioned in close proximity to a conductor carrying a current. The two important conclusions drawn from this observation were that magnetic fields encircle the conductor, and that the strength of the magnetic field is proportional to the applied current (Ørsted's law). Faraday later demonstrated the inverse of Ørsted's law—that changing a magnetic field next to a conductor results in a current being induced in the conductor, and hence formulating electromagnetic induction. Some years later, Siemens developed the first electric power generator based on the discoveries of Faraday. Students are often referred to texts such as [8], [9] to read the history, background, and science of this period, as well as to other feely available resources such as wikipedia.

Much of the work in electromagnetism is concerned with how these forces may be controlled using electric currents, and how they may be modelled mathematically (see for instance [10]). However the inherent complexities of these models make them somewhat daunting and difficult for undergraduate students to grasp if presented as a primarily theoretical subject. A demonstrative, interactive, and experimental approach with reference to real world applications is generally accepted to be more beneficial and likely to lead to the constructivist approach to learning referred to in the previous section.

### A. Induction Heating

One of the largest man-made sources of electromagnetism may be found in electric power transformers. The principle of a transformer is that an alternating current in one winding creates an alternating electromagnetic field and hence induces a current into a secondary winding. However the windings are often wound around a common iron core for better flux linkage and the ac currents can be induced in the iron just as there are currents induced in the windings from the alternating magnetic field. These induced currents (known as eddy currents) circulate through the cross-section of the core. The iron offers a higher resistance to these eddy currents than other conductors, and extra power is dissipated in the form of heat. This heat dissipation side effect is so pronounced that it has been adopted as a means of heating ferrous materials.

### B. The Elements of an Induction Heater

There are three main elements to create an induction heater—a work coil to produce the strong magnetic fields, an electrical conductor to be heated, and a source of high frequency AC power. Figure 1 shows the basic structure of a variable frequency induction heater.

Parallel-LC circuits (also known as tank circuits) are used to store energy. They take energy from a power source, store it alternately in the inductor (L) and the capacitor (C), and produce a continuous AC wave output. The Parallel-LC circuit in a induction heater is depicted in Figure 1. This circuit operates as follows: firstly a power source is applied to the circuit, and the capacitor charges up with potential energy. The power source is then disconnected and the capacitor discharges it's voltage across the inductor. This current flowing through the inductor creates a magnetic field. When the capacitor energy is depleted, the inductors magnetic field starts to collapse, inducing a back EMF—this back EMF causes the energy from the inductor to charge up the capacitor in the reverse direction. Once charged up, the capacitor feeds energy back into the inductor from the opposite direction—thereby restarting the process of creating a magnetic field, subsequent collapse, and creation of a back EMF in the inductor (albeit in opposite polarity to the initial cycle). The complete cycle then restarts from the beginning. If the L and C components were ideal (zero series resistance) they could continue this cycle indefinitely without dampening of the waveform. However this is never the case and an external power source is required to compensate the resistive losses.

Capacitors and inductors are known as reactive components as they oppose the rate of change of voltage and current respectively, and their reactive value changes depending on the frequency of the applied power source. Specifically, as the source frequency increase, capacitor reactance decreases and inductor reactance increases (and vica-versa when source frequency decreases). The term "resonant frequency" refers to the situation where the source frequency is at a value that causes both the inductor and capacitor reactance to equal each other. For a given source voltage, the current oscillating between the reactive components will be strongest at the resonant frequency of the circuit—at resonance the circulating current may be several hundred times larger than the supply current. This amplification of current in the work coil results in a very strong electromagnetic field.

The work coil in the tank circuit of Figure 1 is analogous to a multi-turn primary winding of a transformer, whereas the work piece (the metal to be heated) is equivalent to a single turn secondary winding. The consequence of this is that strong magnetic fields generated in the primary induce strong eddy currents in the work piece. Additionally, the high frequency used in induction heating power supplies gives rise to a phenomenon known as skin effect. The skin effect forces alternating currents to flow in a thin layer towards the surface of the work piece. The skin effect increases the effective resistance of the metal to the passage of the large current, and so greatly increases the heating effect caused by the current induced in the work piece.

Changing the frequency of the tank circuit and hence the frequency of induced currents can alter the depth of induced current penetration in the work piece. The depth of current

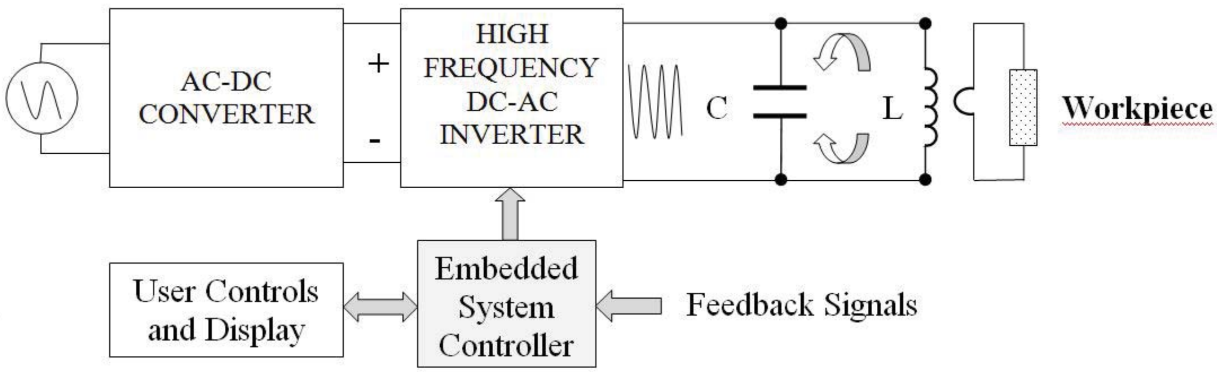



Fig. 1. A variable frequency induction heater

penetration is inversely proportional to frequency—hence a high frequency produces surface heating whilst a low frequency results in through heating and melting. The ratio of skin depth to delta is given in Equation 1, where:

- $\rho$: resitivity of the conductor;
- $\omega$: angular frequency (the rate of change of change of the phase of the sinusoidal wave form) of the current;
- $\mu_r$: relative magnetic permeability of the conductor (the ability of the conductor to support the formation of a magnetic field);
- $\mu_o$ is the permeability of free space (the value of the magnetic permeability in a vacuum).

$$\delta = \sqrt{(\frac{2\rho}{\omega\mu_r\mu_o})} \qquad (1)$$

The intense alternating magnetic field inside the work coil repeatedly magnetises and de-magnetises the iron crystals resulting in friction and additional heating inside the material (Hysteresis loss). Based on this, the work piece used in the induction heater demonstrator presented in this paper is a 3mm diameter iron file that glows red on surface within 15 seconds.

Even at resonance the tank circuit needs a high frequency source of energy to compensate for resistive losses. Figure 2 shows a simple half-bridge DC-AC inverter incorporating two high power semiconductor switches S1 and S2. The circuit operates as follows: S1 is closed while S2 is open thereby delivering current through the tank circuit and charging up C2 to half the dc rail voltage. S1 then opens and S2 closes allowing discharge current from C2 to flow in the opposite direction. The continuation of this cycle results in an alternating current flow in the tank circuit. The rate at which S1 and S2 operate determines the frequency of the inverter. For accurate control, custom designed half bridge driver chips which require a single pulse width modulated (PWM) signal to set the timing and hence frequency of switching are generally used. The PWM signal is the control variable in this case, and can be driven from a central embedded control system that determines the frequency of the tank circuit and hence power to the work piece. The (simplified) design of this embedded control system (for pedagogic purposes) is the primary focus of this paper, and in the following section we present some of our design constraints and choices.

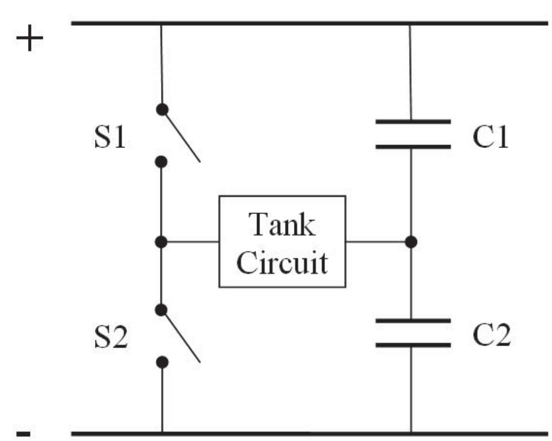



Fig. 2. A half bridge inverter

### C. An induction heater for classroom demonstrations

Commercial induction heaters typically range between 5kW and 500kW, and require three-phase power supplies and pressurised water cooling—this renders them impractical as teaching aids in most academic environments. Smaller induction heaters (of 1kW) are available commercially—however the construction (and IP protection issues) mean that these can only be treated as black boxes and so are equally unsuitable to being used as a teaching aid for first principles.

Although all induction heaters consist of the three main elements presented in Section II-B, the method of power control can vary. The simplest method is to use a fixed resonant tank circuit with variable inverter frequency sourced from an embedded controller. An alternative method is locking the frequency of the inverter and varying the DC rail voltage. The latter method has the advantage of giving 0% to 100% power control however adds more complexity, size and weight in creating a controlled rectifier circuit.

All induction heaters require a form of user interface for control and operation. Typically this consists of a heat on/off and power control along with some form of indicator feedback of the heaters state as well as system fault/ready indicators and system reset facilities. Other useful user feedback and interface facilities include inverter frequency and current, tank volts and current, DC voltage, and output power. Much of these can be implemented in a separate embedded control system with suitable user peripherals if the sophistication of the induction heater requires it. Some advanced commercial models even include facilities for security (such as password control), user programming, and off-board USB and Ethernet connectivity.

The classroom demonstrator presented in this paper was designed with a number of key requirements. The first requirement concerns mobility: needing to be light weight in order to be practically useful for moving around (lecture) campus environments as it was not intended to be restricted to a single laboratory facility. Secondly, the internal structure should be readily visible to the naked eye and to audio-visual lecture aids. Thirdly, user control should be as simple as possible as the focus is on the principles of induction heating rather than on designing complex or advanced induction heaters. Fourthly, it should be able to run using 240v AC, 13 amp 3-pin sockets (as this is the standard power supply in the UK) and not require any form of water cooling. Fifthly, it should heat a work piece to glow red within approximately 10 seconds so as to provide relatively instantaneous control feedback to a classroom. Finally, and most importantly, the software architecture in the control system should be both simple to understand, and flexible to change. Simplicity is important as the demonstrator is used for Engineering students who do not have a comprehensive background in software. Flexibility is important for similar reasons—students should be able to experiment with and alter the control system without needing specialist software skills.

In order to meet software requirements, a time-triggered co-operative (TTC) scheduler running on a COTS micro controller was selected. In the next section, we present the rudimentary principles of a TTC scheduler, and follow this with a description of the software tasks implemented.

## III. Time-triggered Co-operative scheduling

A common definition of the term ”embedded system” is an application that contains at least one programmable computer—typically a micro controller or similar—which is used for some form of systems control by users who may be unaware that the system is computer based. A consequence of this is that one of the challenges in implementing an embedded system is that it is often resource constrained—in terms of processing power, memory, and software resources. It is often the case that a general purpose operating system is not available and it is the responsibility of the embedded system engineer to manage resources (such as scheduling) that would normally be taken care of by an operating system.

The internals of an embedded system are often decomposed into a set of individual items of work that need done—known as tasks. For instance, an embedded system designed to control the flow of water through a turbine may be decomposed into a task that reports the current flow using a sensor, and a task that changes the rate of flow by acting on a valve. While the tasks are inter-related in the sense that they are both needed for the system to operate correctly they are independent in terms of the precise functionality that they offer.

Tasks have to be scheduled and managed—that is to say the order and frequency with which a task needs to run should be managed, along with the resolution of conflict issues such as priority. There is a great deal of literature on many different aspects of scheduling. Broadly speaking, scheduling can be divided into two different techniques: event triggered and time triggered. In the former the embedded system initiates a task based on some external occurrence such as a sensor interrupting with a new report. Consequently it is not always pre-determined which task will run at any given time, or even which task may have priority. In the latter the order and frequency of tasks is predetermined at system design. Run-time of the system is split into a series of time slots, and each task is allocated a slot. As such it is possible to determine at design time which task will run at any point in time. Both approaches have relative advantages and disadvantages. The subject is widely discussed in for instance [11].

In this paper we adopt the approach of time-trigged co-operative scheduling. The term time-triggered means as above: we determine the tasks needed to control the induction heater at design time, and we calculate the schedule needed to support these tasks. The term co-operative means that when a given task is running (i.e, has control of the microprocessor and peripherals) it will always run to completion—neither the scheduler nor other tasks may interrupt it. We further assume that our system design will consist of periodic tasks—tasks that run and re-run over specific pre-determined time intervals.

The reasons for selecting this TTC architecture are pedagogic rather than purely scientific. This architecture forces the design of each periodic task to consider an atomic action—that is to say, any given task performs a single action and has a single point of responsibility to the control system. This decomposition permits students to study each of these points of control individually without having to consider complex software architectures. This facilitates a compositional approach to studying the control system as well as a more tractable piece of software for non-software students to experiment with.

A TTC scheduler has a number of constituent parts—the task array (user tasks) which may be statically or dynamically entered, initialisation and start functions for controlling the scheduler, and a dispatch function for running tasks.

In the initialisation phase, the scheduler adds the user tasks into the task array. This array holds information about each task including the code for the task, and information about the time slots in which the task must run. Then scheduler then enters normal mode—usually implemented as a `while(1)` loop where an on-chip timer and interrupt is used to launch tasks at the correct time. When not actively running tasks, the scheduler may go to sleep or enter power saving modes.

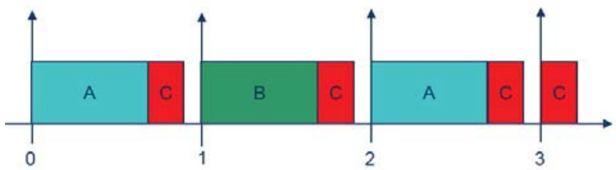



Fig. 3. A timeline of an example task schedule

Figure 3 is a visualisation of a time-triggered co-operative schedule for three tasks A, B, and C. Time increases from left to right with some notional time 0 being the leftmost point. Time is divided into a series of repeating, equidistant slots along this axis and in this example these slots are 10ms apart. As such, every 10ms the scheduler is awoken. On waking up, it checks to see which tasks should be dispatched in that slot and executes them. In this example we assume that all tasks have acceptable execution and jitter times that do not cause them to overrun into an adjacent slot.

Each of the tasks have their relative periods and offsets. For instance, task A has period 2—it is required to run every 2 time slots with an offset of 0 meaning that it runs for the first time in the first time slot. Task B has period 4—it is required to run every 4 time slots; and task C has period 1.

There are many considerations to be made in constructing a TTC schedule, and the interested reader may refer to a number of sources in literature that discuss them such as [12]. In the following section we present the decomposition of tasks necessary to control the demonstrator induction heater and the schedule calculated for them.

## IV. A TASKS SET FOR INDUCTION HEATING CONTROL

The task set consists of eight discreet, periodic tasks. In this section we present a brief description of each tasks—we do not present source code due to space constraints.

1) Tuning frequency update: Timing pulses to the inverter must be maintained at a constant rate and a separate circuit is used to achieve this to allow the micro controller to periodically update the set value. A reference value is set by hand using a potentiometer. The task on the micro controller reads an analogue signal that is used to update the digital timing circuit—this also provides a frequency value for the induction heating task.
2) Heat on timer update: The timer delay potentiometer is read by the micro controller ADC. The task monitors the timer setting dial and updates the IH control task.
3) IH control: The major purpose of this task is to start and stop the heater (achieved using an isolator to the half bridge driver enable/disable pin). However, as a safety precaution it will only start the heater when the start button has been pressed, and the inverter cooling fan is at least 70% of it's rated speed, and the tuning frequency is equal to or greater than the resonant frequency of the LC tank circuit. Similar conditions exist for stopping the heater.
4) Start, stop, reset and lock request scan: This task implements some security and safety features. Start and stop turn the inverter power on and off, whilst reset clears the fault indicators. The lock request disables the start button and requires a pin number to re-enable it.
5) Manual/auto select: This task sets the mode of operation by reading a toggle switch on the control panel. Manual mode allows the user to turn the heater on and off by hand. Auto mode can be used to turn the heater off after a preset time up to 60 seconds.
6) LCD update: The LCD is used to indicate current state of the heater, and is connected by a serial peripheral interface (SPI). It is updated periodically to reflect the latest state of the system. Output information includes the on/off state of the heater, frequency in kHz, fault status, cooling fan speed, manual or auto mode, and preset time delays when in auto mode.
7) Cooling fan monitor: A fan is used to keep the half bridge mosfets cool—and as this is a safety-critical function, it requires constant and accurate monitoring. This task monitors a timed pulse generated by the fan that represents current speed, and this value is passed to the IH control task. If the fan speed drops below 70% overheating is assumed and the inverter is disabled.
8) IH heartbeat: A heartbeat LED task is included to show that the embedded control system is operational and functioning to a user. If the LED is not toggling, it is an indication that an issue with the control system exists.

### A. The schedule

In the previous section, we itemized a description of each of the tasks in the control system. In this section we present a diagrammatic description of the schedule. This description shows task periods, and the relative frequency of each tasks.

Figure 4 show where all eight tasks appear in 500ms major cycle. The most frequent task is the frequency update—run every 10ms to ensure a smooth variable frequency. At 100ms, 200ms, and 500ms other less frequent tasks with different execution times are introduced. The maximum tick interval of 10ms was determined by the sum of all task worst case execution times (WCET) plus room allowing for jitter.

## V. PEDAGOGIC BENEFITS OF THE SOFTWARE ARCHITECTURE

One of the most common classroom activities to demonstrate electromagnetism is studying and using a simple DC motor. A typical experiment that naturally follows on from introducing elementary electromagnetism principles concerns investigating the force exerted on a current carrying conductor in a magnetic field (Flemings Left Hand Rule). This involves attaching a battery to a coil of wire suspended in a permanent magnetic field, resulting in a spinning coil. Whilst useful

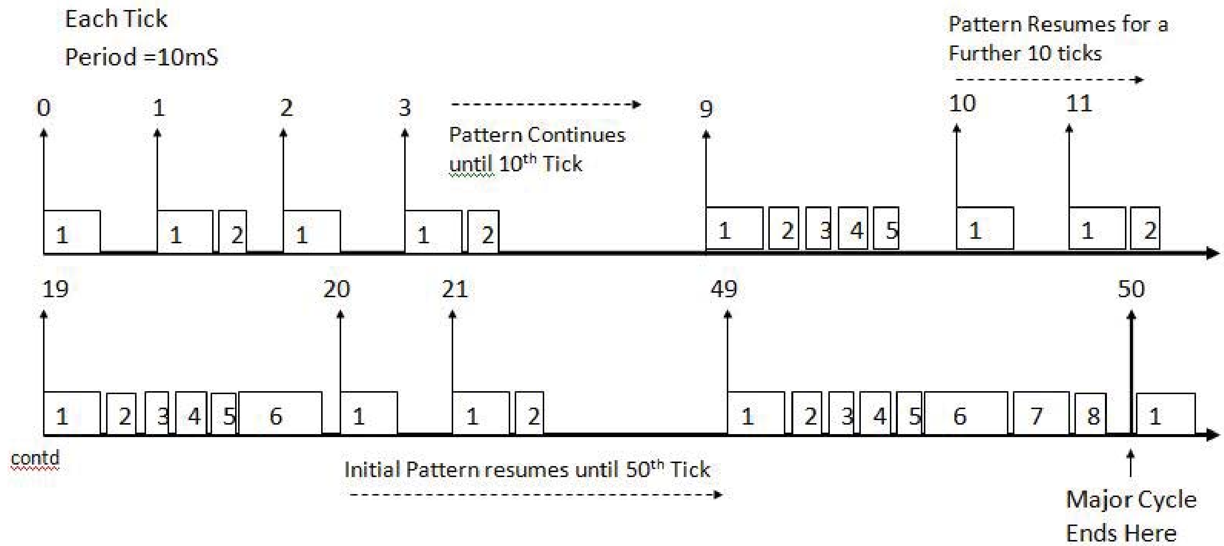



Fig. 4. Major and minor cycles for the task schedule

for introducing first principles this experiment is limited in its applicability to further exploring induction heating and induction heating control due to the inherent difficulty of decomposing the relevant constituent forces.

Despite these inherent difficulties, the combination of electromagnetism, power electronics, electric circuit theory, and control systems in induction heating make it an excellent topic to introduce at this intermediate stage of an undergraduate curriculum as the core scientific principles follow naturally from DC motors. In our experience whilst the topic has been well received by undergraduates at this level, there are two inherent problems. The first concerns practicalities of accessing induction heaters. This involved either having dedicated (medium scale) laboratory facilities for commercial heaters (which restricts class size and therefore experimentation access), or employing field trips to engineering firms (which bring many benefits but also come with the problems of cost, access, and logistics). The second concerns the construction of the induction heater as typically they are manufactured to serve commercial rather than educational purposes.

At University Centre Peterborough in order to manage resource constraints we have typically used student field trips to study and observe commercial induction heaters. Whilst there are benefits to this approach, in addition to some of the limitations above the pragmatics are that follow-up learning and teaching is very theoretical and centred on a transmissive lecture model; it is our belief—generally reflected in literature—that learning is improved if a more experimental approach may be taken.

One such example of a previous student activity was a field trip to a power converters research laboratory. On this trip, the students were introduced to a commercial induction heater which took on the appearance of a large steel cabinet with a front mounted control panel. The induction heater was started and a large piece of metal consequently heated up—but the learning benefits were extremely limited because the internals of the process, and the scientific principles governing those internals were very much obfuscated as the equipment was not designed for learning and teaching purposes.

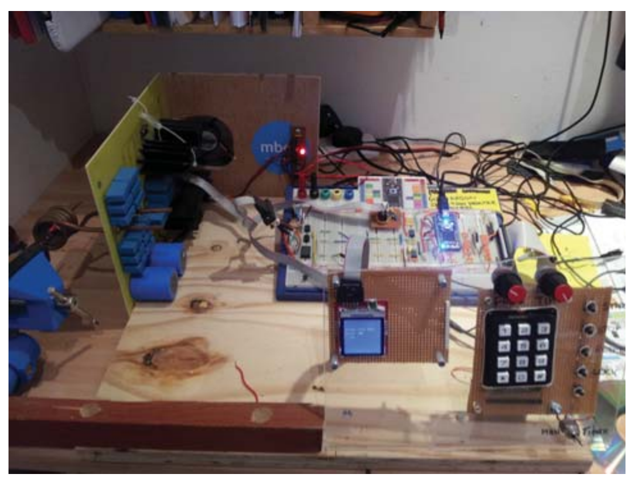


Fig. 5. The prototype demonstrator

We have developed a prototype demonstrator, shown in Figure 5 and used in redesigning the student activities. Instead of employing field trips, the prototype demonstrator is used in a lecture environment with a projector display for large group teaching—and this greatly facilitates the introduction to the subject. The open construction of the unit simplifies the identification and discussion of individual components.

Although we have only built a single demonstrator, the cost is a major success. The cost is very low compared to either a commercial heater or of field trips—the consequence of this is that it is viable to equip a laboratory with a number of these. Moreover as the construction is small and portable the laboratory facility need not be dedicated.

It is this result that enables the most significant change to learning and teaching activities. We are in the process of designing experiments that may be conducted as small group activity—this was not possible with previous resources. Moreover these experiments involve not only investigating the physical construction of the demonstrator but also the software control system. Previously experimentation with a control system proved impractical as either source code was not available, or too complex for non-software students to experiment with. The time-triggered co-operative architecture used in our control system has the benefits of separating all elements of control; and our early experiences suggest that this separation of concerns not only highlights the important aspects of control from a learning perspective but also makes the implementation tractable to students with a limited background in software and programming.

## VI. Conclusions and future work

In this paper we have presented the design of our time-triggered co-operative control software for our induction heating controller. In terms of correctness (and safety) of the implementation we have conducted a significant amount of analysis and testing—however this has not been presented in this paper as it is not the focus. Our demonstrator has been used as a teaching aid on the Undergraduate Engineering programme at University Centre Peterborough. Anecdotally, the demonstrator has been well received by students and in our opinion has made the presentation and analysis of the principles of induction heating much easier. We do not yet have concrete results supporting the pedagogic benefits of our hypothesis (primarily because the cohort of students to which this was introduced have only undergone summative and not formative assessment on the topic at time of writing), but an item of future work is to consider in more depth how we may evidence the hypothesis that adopting the time-triggered co-operative approach to the control system has produced a compositional model that aids student learning. A further item of future work is to consider other control problems discussed on the degree and investigate whether or not these would also benefit from this model of presentation.

Some further improvements to the implementation presented in this paper include using a central controller to do both the general housekeeping tasks described in this paper, as well as phase lock loop control. In this scenario we believe it would be beneficial to use a digital signal processing platform to perform the real-time monitoring of the inverter and tank waveforms. In addition to the architectural benefits of this, there are complementary pedagogic benefits—specifically that capturing the waveforms of the induction heater in this way would enable us to display them on a large projector instead of a small screened oscilloscope. This has the benefit of enabling the equipment to be used in a large lecture theatre environment instead of exclusively in small group teaching.

Another item of future work is to open up the implementation of the control system to further functional improvements—possibly as undergraduate projects. This is of interest because the task of building or enhancing induction heating control systems is one that has previously been challenging to support at undergraduate project level. To this end improvements could include, for instance, directly accessing peripherals on the development board instead of using IDE library support. Another project could be to investigate the effects of jitter in the scheduler (a common and well-understood problematic phenomenon of time-triggered schedulers) and any resultant detrimental effects on the heater performance.

When ferromagnetic materials such as iron are heated, at a certain temperature their magnetic properties change (Curie temperature) hence changing the inductive value and leading to a shift in resonant frequency of the tank circuit. To keep maximum power to the load the inverter frequency has to be adjusted manually. At resonance the inverter voltage leads the tank capacitor voltage by 90 degrees so by introducing a phase locked loop control system the inverter frequency could be made to automatically track the changing resonant frequency of the tank circuit. COTS chips such as the 4066 that can monitor the inverter and tank voltages phase difference. If there is a phase shift either above or below 90 degrees an error signal will adjust the voltage controlled oscillator output to bring the tank circuit back into resonance and will continue to do so—and so another undergraduate project could be to port the current systems to this new hardware, and measure the functional benefits resulting thereof.

## References

[1] M. Sibley. *Introduction to Electromagnetism*. Elsevier.
[2] W. R. Phillips I. S. Grant. *Electromagnetism*, volume The Manchester physics series. Wiley, 1990.
[3] Stephen J. Blundell. *Magnetism: A Very Short Introduction*. Oxford University Press, 2012.
[4] Lee Billings. A rocket for the 21st century, September 2009.
[5] Y. J. Dori and J. W. Belcher. How does technology-enabled active learning affect undergraduate students understanding of electromagnetism concepts. *The journal of the learning sciences*, 2005.
[6] J. G. M. M. Smits J. C. Perrenet, P. A. J. Bouhuijs. The suitability of problem-based learning for engineering education: Theory and practice. *Teaching in higher education*, 2010.
[7] Jalal Jalali Frank Li, Theodore Burke. Developing and induction heating system laboratory with dsp microprocessors and power electronic devices.
[8] Alan W. Hirshfield. *The Electric Life of Michael Faraday*. Walker books, 2006.
[9] Walter C. Patterson. *Transforming Electricity: The Coming Generation of Change*. Routledge, 1999.
[10] J. N. Ross P. R. Wilson and A. D. Brown. Simulation of magnetic component models in electric circuits including dynamic thermal effects. *IEEE Transactions on Power Electronics*, 2002.
[11] H. Kopetz. Time-triggered real-time computing. *Annual reviews in control*, 2003.
[12] A. Burns. Scheduling hard real-time systems: a review. 1991.

# Bridging the Education/ICT-Business competency gaps: ontology based solutions

**A. Zainal[1], and A. Bouras[1,2]**
[1]Department of Computer Science and Engineering, College of Engineering, Qatar University, Doha, Qatar
[2]IctQATAR Chair, Ministry of Information and Communications Technology, Doha, Qatar

a.zainal@qu.edu.qa, abdelaziz.bouras@qu.edu.qa

**Abstract-** The use of ontologies is a key step forward for describing the semantics of information on the web. It is becoming more and more important to make the information machine readable, since the volume of data is continuously growing. In this paper we deal with the use of such ontologies based solutions that would help in analyzing and understanding the gap between competencies needed for occupations offered by the industry and learning outcomes offered by academic programs. The higher education sector is one of the main suppliers of work force who feeds the working industry. This dependability of the industry on higher education adds pressure in how academic programs are constructed, constantly trying to meet the ever changing industry requirements for competencies. Can the base ontologies contain sufficient information and knowledge that would help understanding the relation between learning outcomes and the collection of skills? Can they clearly describe the set of skills, abilities and knowledge required in individual job occupations? The paper addresses such questions through a simple reference scenario to open discussion.

**Keywords**: Ontology, OWL, Competency, Education.

## 1. Introduction

Employees' qualifications could potentially be one of the main reasons for scoring a job. One's education background would primarily represent the knowledge the individual possesses and shapes his expertise. After education, qualification, skills, etc. and other criteria are taken into consideration when being considered for hiring our current industry.

Educators struggle to find an accurate mechanism that allows them to understand the gaps between the learning outcomes of their programs and the competencies required for the jobs they target to cater to.

On the other spectrum, Industry employers may not be fully aware of the new learning outcomes that are generated by existing programs that could fit to the jobs they offer. They may not be also aware of new programs emerging and how can they fit different types of jobs. Understanding the employee's educational background may be a key factor in effectively identifying what training programs are needed for the employee in his/her first few year of employment. The analysis could potentially save the company's money by avoiding the offer of training programs that generate competencies that already was possessed by the employee.

Understanding what the industry requires and what opportunities are available for certain majors could also possibly be a good source for career guidance to students.

By identifying the courses the student enrolled in during the academic program, he/she can choose a certain job title and identify the gap between the competencies required and what he/she currently possesses. The used ontologies will allow higher educators to see the list of learning outcomes that their programs do not generate in order to help them in their reviewing the existing and planning the their new programs.

## 2. Current developments based on ontologies

Ontologies developed in the Education domain tend to focus on modeling courses, students, learning outcomes and competencies [12]. Some examples of these ontologies are: Course Ontology [2], ALOCoM[6] and AcademIS [11].

Common problems in designing an Ontology in this domain are the means to maintain them, where most developed ontologies suffer from this at some point through the Ontology development phases [12]. Ontologies related to Education are also highly being used for E-Learning, where they are used to represent: curriculum modeling and managing, learning domains, Students, and Services offered by E-Learning facilities [1]. Systems and frameworks are developed on top of these ontologies to help answer critical questions asked by the users of the domain, such as: OntoEdu [4] and CUBER project [9].

In the Education domain, few efforts were noticed that integrated the representation of industry/Employers description and its relation to the Education domain. One of the examples found is an ICT field specific ontology called "ICT Education Otology" [3].

On the other spectrum, the domain of Industry had efforts to construct ontologies for Competencies, in order for employers to best understand how to provide effective Human Recourse training programs [10].

It is also noted that few efforts have linked the working Competencies to educational competencies generated by the learning outcomes of study programs in the Education domain.

## 3. Competency Reference Model design

Chosen languages to model the ontologies generally belong to the semantic web family. We have here based our reasoning and developments choosing the OWL language, and utilizing the Protégé tool for implementation.

This section provides details about the chosen methodology of constructing the Competency Reference Ontology (CRO). This ontology will be the base of the Competency Reference Model design.

After surveying the existing methodologies of constructing ontologies, it was decided that the followed methodology will be used to build the CRO; "Ontology Development 101" by Natalya F. Noy and Deborah L. McGuinness [8].

The basic concept of the methodology is to start with an initial draft of the ontology and then revise and remodel the ontology as you go on. This will be iterated until all the details are filled and the desired representation of the domain of knowledge is satisfied. The approach suggests repeating the designed steps iteratively until reaching the final desirable ontology design.

Design decisions are guided by some fundamental rules outlined in the paper [8], the rules emphasizes that: There is no sole optimal way to model a domain, this is dependent on the application and the anticipated extensions. Also, it restates that developing the ontology is inevitably an iterative process and that the represented objects and relationships can be compared to nouns and verbs used to describe the domain of interest in the ontology.

### 3.1 Ontology construction steps

#### 3.1.1 Determine the domain and scope of the ontology

Determining the domain and scope of the ontology was achieved by answering the set of questions listed in this step. The questions and their answers for the CRO is listed in Table (1).

Table 1. Questions and answers to determine the Ontology domain and scope

| Question | Answer |
|---|---|
| What is the domain that the ontology will cover? | Education domain and the Industry domain. |
| For what we are going to use the ontology? | To perform a gap analysis between the two domains to help employers in the industry and educators from academic institutions understand where do they stand. Academic intuitions will be much more aware of the current demand for competencies and specifically improve the design of their programs to increase the quality to the students they supply to the industry. On the other hand, this will make employers more aware of the available competencies that could be utilized and fit different job roles. |
| For what types of questions the information in the ontology should provide answers? | • Graduates of which program have the competencies needed for this particular job description (a collection of competencies)?<br>• What competency is missing from students profile graduating from a particular program? |
| Who will use and maintain the ontology? | Maintained by users from the academic side, individuals must be updated manually for all classes in order for the ontology to be timely accurate. |

### 3.1.2 Consider reusing existing ontologies

Several ontologies were found in the two domains, however, no ontology could have been re-used as it is to serve the intended purpose of the design. The existing ontologies has inspired the direction of the new ontology and the way terminologies were refined to better represent the domain.

### 3.1.3 Enumerate important terms in the ontology

The ontology was represented in a taxonomy that helped in describing employee, education and industry defined competencies. The Superclass was further divided into subclasses that helped in defining more classification to the individuals.

All classes under Thing were set to be disjoint, this implies that an individual cannot be part of two classes at the same time. An example would be that a single course cannot be classified under the class Course and Learning Outcomes at the same time, it can only be a member of one of them.

Some of the main terms that were listed to represent the education domain were: Institution, Department, Course, Program, Learning Outcome, Grade and Study Plan". Terms representing the industry domain included: Employee, Competencies, and Occupation.

### 3.1.4. Define the classes and the class hierarchy

Classes are groups of individuals that are chosen to represent a class because they fulfill the same membership requirements as others [5]. Classes usually exist in hierarchy often referred to as Taxonomy. Hierarchy is used to infer inheritance, this allows the *Reasoners* to function and do their job [5].

Defining the classes was done in a *Combination development* process, which is using both top-to-Bottom and Bottom-to-Top approaches. Prominent terms were first coined and then followed more generalization and specialization that created the hierarchy shown in Figure (1).

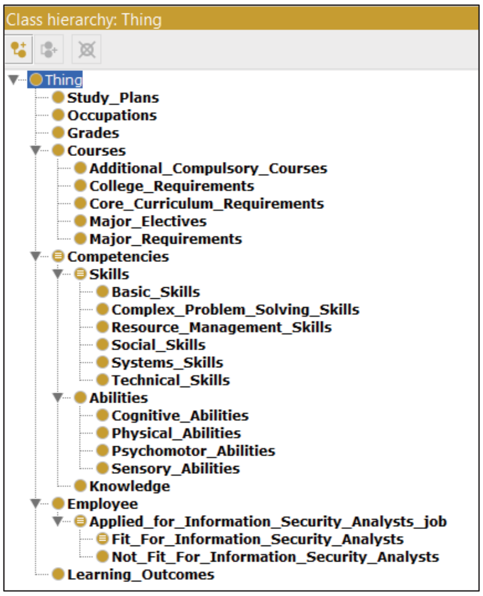



*Figure 1 CRO class hierarchy*

Most of the classes that were derived from the Education domain have remained general as noticed like: *Study_Plans, Grades and Learning_Outcomes. Courses* has been categorized into more subclasses to mimic the true classification nature as represented in the domain. As for classes that were derived from the Industry domain, classes such as *Competencies*, has been classified according to O*NET's [7] native classification of skill, abilities and knowledge. The *Employee* class will have a subclass created for every job occupation that the employee would like to test the gap analysis on. These classes will be used by the *Reasoner* to infer how fit are the *Employees* whom have applied to the jobs they seek.

### 3.1.5. Define the properties of classes—slots

*Properties* will typically come from the verbs we use in the domain to describe the classes. Some of the verbs that would describe the enumerated terms in step 3 are: *Enrolled, generate, has applied, has course, has gained, has selected, is a, is equivalent to, is part of, is selected by and requires. Properties* serves the purpose of linking two individuals together, thus, each *slot* mentioned was used to describe these links to create the internal concept structure. The defined *Object Properties* of the ontology is as listed in Figure (2). There were no defined *Data Properties* yet for CRO, as there has not been any need to use them in the model.

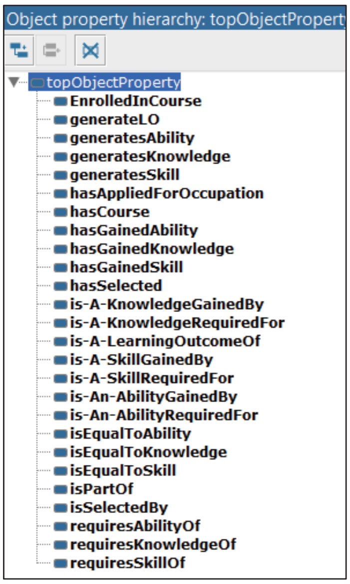



*Figure 2 List of Object Properties used in CRO*

### 3.1.6. Define the facets of the slots

Defining *Properties (Slots) facets* can help in describing several features about the *slot*, like, value types or number of values. The type of *facet* that was used in the ontology was *Domain* and *Range Slots*, where every *Property* has a designated *Domain* and *Range* to restrict the inference results as shown in Figure (3).

Some of the *facets* on *Classes* are introduced in the ontology in Table (2). The logic behind the first facet is to restrict the membership of the class to include all *Employees* who are intending to apply for the specific job mentioned in the class.

The way the *facet* expression was written, ensures that any Individual under the Class

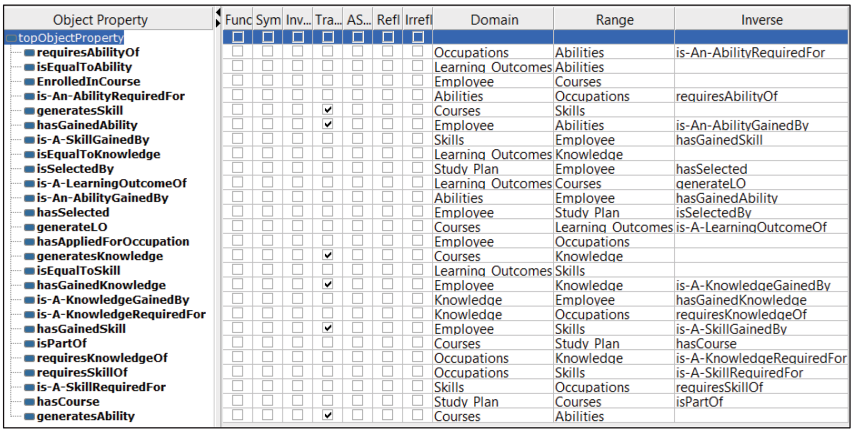



*Figure 3 CRO Domain and Range Slots matrix*

Employee who has the Property Assertion "hasAppliedForOccupation value Information_Security_Analysts", gets classified by the *Reasoner* under this class.

The second facet will classify an *Employee* as "Fit for the job" if he/she has gained all the *Skills*, *Knowledge* and *Abilities* required by the intended occupation. Gaining the mentioned competencies are not explicitly asserted by the user, however, inferred by the *Reasoner* from the *Learning outcomes* generated by course work undertaken by the student. Figure (4) shows the inferred Competencies for Noor.

In order for achieve this inference, a *SubProperty Chain* was added to the *Property generatesSkill* as shown in Figure (5).

*Table 2. Class and Facets used in CRO*

| Class Name | Facet |
|---|---|
| Applied_for_Information_Security_Analysts_job | Employee and (hasAppliedForOccupation value Information_Security_Analysts) |
| Fit_for_Information_Security_Analysts_job | (hasGainedAbility only (is-An-AbilityRequiredFor value Information_Security_Analysts)) and (hasGainedKnowledge only (is-A-KnowledgeRequiredFor value Information_Security_Analysts)) and (hasGainedSkill only (is-A-SkillRequiredFor value Information_Security_Analysts)) |

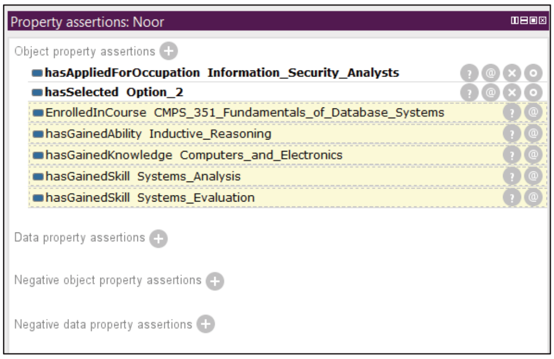



*Figure 4 Example of inferred Competencies for Noor*

### 3.1.7. Create instances

The ontology creation was concluded by adding the required *individuals (instances)* to the *classes* of the hierarchy. The process requires choosing a certain *class*, adding the *individual* and then appending it to the necessary *slot* values or in other words, asserting the *Property* to the *Individual*. For example, as shown in Figure (6), *Noor* is added as an *Individual* of the *Class Employee*. She has two Object Properties asserted to her, the first *Property* is *hasAppliedForOccupation* and it is asserted to an *Instant* from the *Occupation Class* which is *Information_Security_Analysts*. This implies that *Noor* has applied for the asserted job title. The second *Property hasSelected* has been asserted the value *Option_2*, which denotes Noor's selection of that Study Plan.

Due to the nature of Open World Assumptions (OWA) OWL has, individuals are assumed to be the same regardless of the way they are named [5]. Two individuals may have the same name but they could be assumed to be different. Likewise, when two individuals may have different names and could be assumed to be equivalent. This requires us to explicitly define all the other individuals in the Class *Employee* as different Individuals. This will drive the *Reasoner* to not assume that other individual are equal and would prevent inconsistent inheritance.

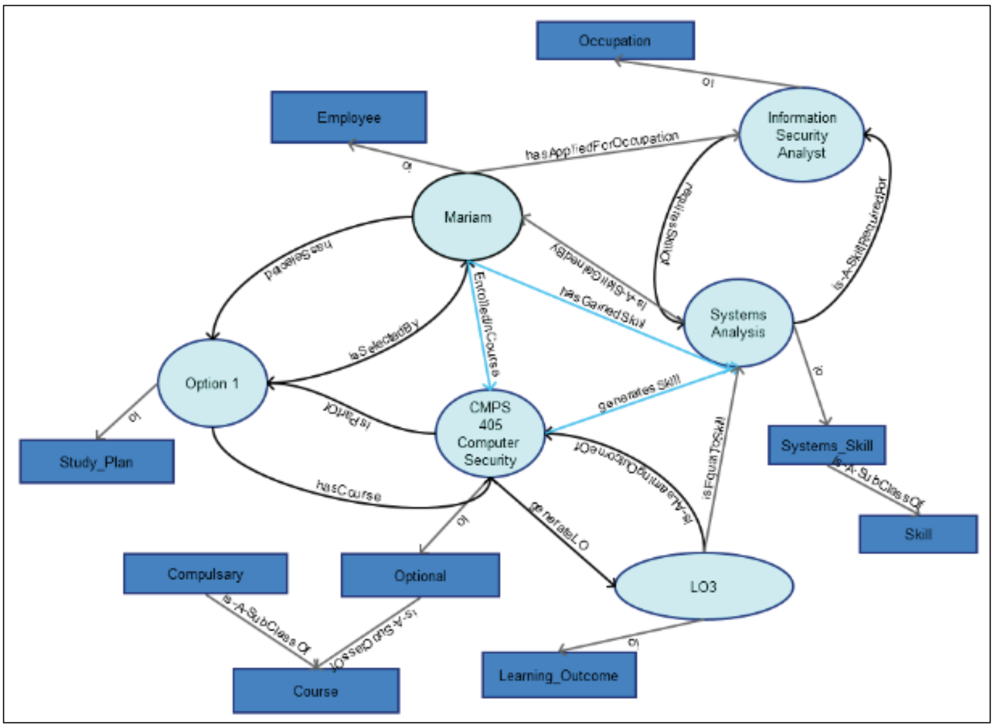



*Figure 4 CRO Class and Property Diagram*

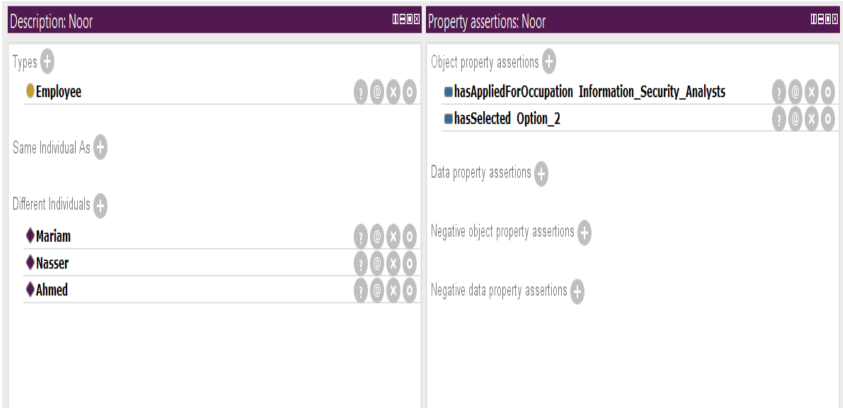



Figure 5 an example of Individual Object property assertion

**3.2 System architecture**

The system offers an interface for different users: Students, Employers and Educators. The designed ontology will be the base of the system, other data repositories such as O*Net and University Course and learning outcome databases can be plugged into the system to supply the ontology with the necessary class individuals. Both O*Net and the University databases (Figure 7) will be needed to be maintained manually by the system admin. Future work could include text-learning ontologies on both databases to derive the automated mechanism of maintaining the system.

Example of system uses would be, a student user will be asked to input the desired job occupation he seeks to apply to. The system would ask the student to enter his study plan by supplying a list of courses offered by the program he enrolled in, where he needs to select them. The data inputted in the user forms will be plugged into the ontology and asserted to the related classes, properties and individuals. The *Reasoner* will be run and the desired query result will be displayed to the user. A student could be classified under "Fit" or "Unfit" to the chosen occupation. In case of being "unfit", another query is run by the ontology to compare the competencies gained from the education programs and the competencies required from the selected job. The competencies that does not belong to the two sets will appear as a list of competencies that the student is lacking. This is intended to provide guidance to students on what it takes to be fit for the selected job.

Another use of the system would be by Educators. When comparing the learning outcomes of a model study plan in computer science Undergraduate program to the competencies required in all ICT jobs existing in O*Net. The ontology would be able to answer different queries such as: what is the optimal study plan (Collection of courses) for the job *Information Security Analyst*? what competencies the study plans lacks to generate outcomes that are comparable to current ICT jobs? etc.

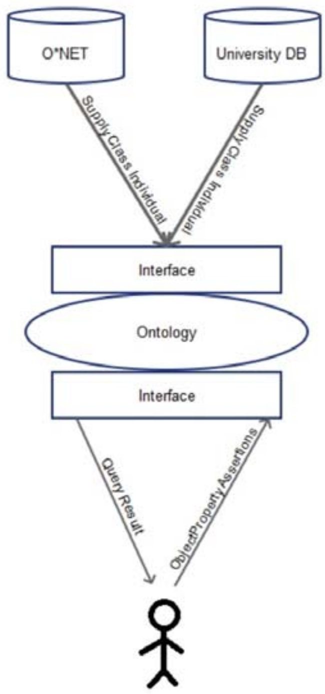



Figure 7 Use Case diagram for the Competency Reference Model

## 4. Evaluation and Testing

At this stage, we are still testing and refining the ontology design. It has been noted that with the current ontology design, *Noor* would be classified as "Fit" for the chosen occupation if she has gained exactly all competencies required for the occupation she applied for. This achieves what the ontology is meant for, since the set of Competencies gained by the Employee is equal to the set of Competencies the Occupation applied for requires. However, the ontology will fail to classify an Employee as "Fit" if he/she has a set of competencies gained that are more than what is required by the set of competencies required for the job. The logic currently assumes that if the two competencies sets are exactly equal, then the condition is satisfied. However, the correct logic would be that, if the set of Competencies required by the occupation is a Subset of Competencies gained by the Employee then the condition should be satisfied. We are currently refining the design of the ontology in order to achieve that desired expression, though, due to the restrictive nature of OWL DL, this has proven to be a challenging task.

## 5. Discussion

In the educational and industry areas, high level data are considered to be helpful and ontologies can be used as a mean for formal representation of their knowledge domain. The use of underlying formal languages such OWL improve the retrieval and interchange of educational resources. Ontology-based models exhibit characteristics such extensibility, ease of sharing and reuse, and logic reasoning support. However some questions still remain open. For instance, how to overcome limitations on expressivity? How to deal with noise such as synonymy, polysemy, homonymy, multilingualism, etc. How to be sure that the represented knowledge is adequate and satisfies the goal? How to deal with quality of knowledge within the knowledge representation? How much consistent the represented knowledge is using ontologies? The authors will be happy to discuss the details of these questions during the conference.

**ACKNOWLEDGMENT**
This publication was made possible by NPRP grant #7-1883-5-289 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

## 6. References

[1] Al-Yahya, M., George, R., Alfaries, A. (2015). Ontologies in E-Learning: Review of the Literature. Int. Journal of Software Engineering and Its Applications, 9(2), 67-84.

[2] Ameen, A., Khan, K. U. R., Rani, B. P. (2012, July). Creation of Ontology in Education Domain. In T4E (pp. 237-238).

[3] Chin, K. L., Chang, E. (2011, May). A sustainable ICT education ontology. In 5th IEEE Int. Conf. on Digital Ecosystems and Technologies (DEST), (pp. 350-354).

[4] Guangzuo, C., Fei, C., Hu, C., Shufang, L. (2004, Apr.). OntoEdu: a case study of ontology-based education grid system for e-learning. GCCCE2004 Int. conf. Hong Kong.

[5] Horridge, M., Jupp, S., Moulton, G., Rector, A., Stevens, R., Wroe, C. (2009). A Practical Guide To Building OWL Ontologies Using Protégé 4 and CO-ODE Tools Edition1. 2. University of Manchester.

[6] Knight, C., Gašević, D., Richards, G. (2005). Ontologies to integrate learning design and learning content. Journal of Interactive Media in Education, 2005(1), Art-6.

[7] National Center for O*NET Development. O*NET OnLine. Retrieved May 2015: http://www.onetonline.org/.

[8] Noy, N., & McGuinness, D. L. (2001). Ontology development 101. Knowledge Systems Laboratory, Stanford University.

[9] Poyry, P., Puustjarvi, J. (2003, Jul.). CUBER: A personalised curriculum builder. 3rd IEEE Int. Conf. on Advanced Learning Technologies, pp. 326-327.

[10] Schmidt, A., Kunzmann, C. (2006, January). Towards a human resource development ontology for combining competence management and technology-enhanced workplace learning. OTM Workshops On the Move to Meaningful Internet Systems, pp. 1078-1087. Springer.

[11] Triperina, Evangelia, Cleo Sgouropoulou, Anastasios Tsolakidis (2013). AcademIS: an ontology for representing academic activity and collaborations within HEIs. 17th ACM Panhellenic Conference on Informatics..

[12] Winter, M., Brooks, C. A., Greer, J. E. (2005, May). Towards Best Practices for Semantic Web Student Modelling. In AIED, pp. 694-701.

# Using Bottom-up Techniques in a Digital Hardware Design Laboratory to Better Facilitate Experiential Learning

**B. Earl Wells**
Department of Electrical and Computer Engineering, University of Alabama in Huntsville,
Huntsville, Alabama, U.S.A.

**Abstract -** *This paper presents the core elements of a laboratory assignment that was given in the Spring 2015 semester to a Computer Engineering logic design class. Unlike the other assignments in this class a bottom-up approach to learning was emphasized. Students were taken through the steps of creating the main subcomponents of their design using a combination of first principles and publicly accessible information. After evaluating the correctness of the base design students were shown how new designs could be created by making minor modifications to subcomponents of their original design. Specifically students utilized Pulse Density Modulation to create a sinusoidal waveform generator. This design was then modified to output prerecorded audio to a speaker. The final design was created by combining the first and second designs to create a new design that was capable of transmitting this audio for short distances wirelessly by amplitude modulating a 1Mhz signal.*

**Keywords**: Digital Logic Design, Experiential Learning, Bottom-up Design, Pulse Density Modulation, Verilog HDL

## 1 Introduction

Experiential learning occurs when students take an active role to apply in a trial-and-error manner the concepts they are learning to solve relevant problems[1]. This is the essence of engineering and the applied sciences and supports the primary way that many people learn. Confucius is reported to have said, “I hear and I forget, I see and I remember, I do and I understand”. Experiential learning opportunities in the Computer Engineering curriculum at UAH are centered around in-class demonstrations and projects, homework assignments, simulation assignments, laboratory assignments, and a comprehensive senior design experience. This paper focuses upon how the experiential learning experience can be enhanced in an intermediate digital logic design laboratory.

In general, laboratory experiences in Computer Engineering at UAH differ from simulations and homeworks by their overall larger scope, need for specialized equipment, and their often real-time interaction with external equipment, human interfaces, or with the local physical environment within the laboratory. While a digital design laboratory may employ different kinds of simulations as part of the overall design flow the ultimate goal of this experience has usually been to create real-world prototype implementations of the actual design using Field Programmable Gate Array, FPGA, technology. This allows students to experience the fruits of their labor first hand and to discern the merits and limitations of simulation and other post design analysis approaches in an empirical manner.

### 1.1 Past Digital Design Experiences

Approximately 18 years ago the National Science Foundation provided the seed funding for the digital design laboratory which was at that time called the Rapid Prototyping Laboratory[2-3]. The goals of this initiative were to create a laboratory that could be synchronized pedagogically with the digital design class to provide a just-in-time hands-on learning experience. The laboratory was to incorporate the integrated set of Computer Aided Design, CAD, design entry, simulation, and synthesis tools to be applied to rapidly prototype digital circuits. The laboratory has been very successful and has benefited hundreds of Computer Engineering and Electrical Engineering students since its inception but as the background of our undergraduate students has changed so has the focus of the laboratory assignments.

For example, in the early days of this lab students were given relatively simple but open-ended type laboratory assignments. This worked well because these students were generally positively-motivated by how quickly digital circuitry could be placed in the FPGA by the CAD tool software. Their reference point though was that of the differences in the time associated with CAD tool design entry and synthesis and the time it took them to develop a paper design and manually construct the design by wiring it on a solderless protoboard using discrete small-scale and medium scale integrated circuits.

Within about five years after the establishment of the laboratory, though, students started to become frustrated by how long it took them to synthesize digital circuits compared to compiling standard instruction set code for embedded microcontrollers. Unlike the previous students these students had very little experience with manual prototyping methods. They also had a hard time getting excited by the end product that was to be created. They had already been exposed through video games and other electronic venues to applications that were much more interesting than lighting up LEDs to achieve certain Boolean functions!

To combat this problem more complex but also much more structured laboratory assignments were created. These labs were presented in a top-down manner using the standard functional decomposition design methodology with the top-level often being expressed structurally in the Hardware Description Language, HDL. To handle the complexity the model for some of the components would be provided for the students as Intellectual Property, IP core elements. Students then were to focus on the detailed design of one or more of the other components to which they were not provided IP core models. When the design for these elements had been developed (and they had passed their unit tests) then the students would be able to observe the full functionality of the overall system once their components were successfully included in the design. The reward of their detailed design effort was to have a complete working system. In this manner students were exposed to the important concepts of design reuse as well as system integration issues which are very prevalent in real world design scenarios. This allowed more complex and appealing laboratory assignments which have included such assignments as temperature sensors, capacitance testers, video display/image processing, single button texters, keypad/keyboard interfaces, and analog/digital interfaces.

### 1.2 Weaknesses of the Current Paradigm

One of the issues associated with this top-down approach though is that there is little incentive for students to take the time to examine the internal construction of the completed IP core subcomponent modules even though the full HDL source code for these models was given to them. More often than not students would fail to look beyond the design interface section of the model for any external IP core element. This was unfortunate since there was much to be gleaned by examining how the modules were created. These IP core elements often served as good examples of how various aspects of the hardware description language could be used to model a wide range of hardware configurations. In addition the style and structure of the HDL code contained in these modules could often be easily extended and applied within the other subcomponents that the students were tasked to provided.

While information hiding is certainly an important property of functional, object-oriented, and model-based design it is also very important that the designers understand as much as possible about the functionality of each subcomponent module that makes up such a system. The extreme worst case scenario is that the external IP core modules themselves are viewed as traditional black boxes and there is no real knowledge of what is contained within the modules themselves. Of course something concerning the functionality of the IP module <u>has</u> to be known before it is placed in the design. The more that is known though the better. Such questions as what is the purpose of the module? What algorithms are being used to accomplish the purpose of the module? What are the limitations of these algorithms? What are the data ranges to be expected? How was the functionality of the module been verified and validated? One way that one gains a fuller understanding of the required functionality of a module is by having experience creating similar modules in the past. By providing predeveloped IP to the students for some of the most complex elements we have denied them this experience.

A more troubling issue, though, was the apparent belief by many of the students that since the IP core modules were provided to them by the instructor they must be too complex for them to understand or to create themselves. In other words, some students appear to only be comfortable working in environments that focused on the higher levels of abstraction and thereby became dependent upon others to create IP cores that they could link together in an appropriate manner to create a system level design. This belief appears to be rampant with today's college age students who have had much success programming and interacting with a large number of computer and cell phone type apps that have insulated them from lower levels of abstraction.

While there is much to be said for working at the highest level of abstraction possible it is also extremely important for computer engineers to be able to transverse down into the design hierarchy and develop new lower-level modules when such an action is required. This skill becomes more important for performance and resource critical systems. There should also be no artificial mystique associated with pre-existing modules. There is nothing magic about the IP core modules. They were created using a similar creative thought process and using similar design methodologies as any other element that makes up the system.

### 1.3 Benefits of the Bottom-up approach

This paper was written to describe an effort to overcome some of these wrong perceptions that have been occurring and to foster a more meaningful learning experience. The idea was to develop a new laboratory assignment to augment the existing set of assignments that would use bottom-up design concepts to create a set of useful subcomponents whose functionality could first be understood by the student. After which these subcomponents could be combined in different ways and altered as necessary to create new and more complex systems.

### 1.4 Outline of Paper

The remaining portion of this paper is organized as follows. Section 2 givens an overview of how the topic area of the specific laboratory assignment was selected. Section 3 outlines the laboratory assignment itself, and Section 4 discusses the preliminary results and conclusions that relate to the introduction of this assignment.

## 2 Assignment Selection Criteria

An undesirable side effect of creating any form of structured laboratories is that students are denied the opportunity to think critically in regards to the selection of the design problem to be solved and are also often constrained considerably on how the problem is to be solved. This is also the case for the assignment discussed in this paper but to help counter this lack of open-endedness the criteria that was used by the author in creating this assignment was discussed extensively in class. Even though the criteria for lab creation was based upon educational concerns there is a similarity in many regards to the resulting requirements that come from the needs of the customer in an engineering design scenario. The requirements for this design experience are summarized below:

- The design problem should be highly unique and relevant to today's students.
- It should require the use of general principles that are transportable to other designs.
- The design problem should allow for the accumulation of knowledge to occur throughout the design experience.
- Students should be involved in the design and implementation of every component module in the system.
- The design problem should be composed of two or more phases where each phase builds upon the results of the previous one.
- The design problem should benefit from the run-time efficiency of dedicated digital hardware.
- The resulting design should become more interesting as the design process progresses.
- The design should express some degree of out-of-the-box thinking.

Students were then informed of the overall process that was used by the author of this paper to determine the scope of the project. A topic that seemed to meet the requirements was something in the general area of creating a digitally recorded audio playback system. The author had previously become aware that devices that had direct earphone connections such as mp3 players and smart phones drove these earphones using their digital ports instead of employing traditional digital to analog converters. It was not known, though, by him how this was accomplished. Based upon his limited knowledge of digital signal processing techniques he postulated that some form of pulse width modulation was used to accomplish this task. He then performed an internet search using a standard search engine and found that pulse width modulation was a more appropriate technique for such applications as industrial motor control and the technique that should be employed for audio should instead should probably be Pulse Density Modulation, PDM. Pseudo code for the base PDM scheme was available on Wikipedia[4]. This code though was designed more to illustrate the technique in a non-real time environment such as MatLab. It had to be restructured to be put it in a form that only used scalars and could be easily implemented in hardware. The evaluation of the restructured C/C++ version was then compared to the known outputs that were reported by Wikipedia. To meet the requirement for maximum run-time efficiency fixed point was chosen instead of floating point. (The use of floating point would have also violated either the requirement for no external IP core elements or drastically increased the scope of the problem.)

During laboratory development the author formulated a simple sine wave test to verify the functionality of the audio playback mechanism. This was recognized as an interesting result in its own right and was later made to be the first phase of the laboratory. An unknown element that was not determined until the reference design was actually implemented was would the ambient filtering characteristics of the protoboard and the solderless breadboard act enough like a low pass filter to allow for a measurable waveform to appear on the oscilloscope? It was also unclear before the implementation if it would be possible to clock the PDM waveform at the desired rate of 50 Mhz because of the parasitic capacitances of the experimental setup. These were verified empirically during the sine test.

After the sine wave tests were complete then it was a simple matter to incorporate the original digital playback idea. This required creating modules that could drive the PDM component at an appropriate sample rate to output prerecorded audio which was taken from public domain sources on the internet. This rate should be twice the maximum desirable frequency. To allow for maximum non-compressed audio storage within the FPGA a sample rate of 8K was chosen.

The final phase of the lab was created by the author after observing that the sine wave and the audio generating sections could be very easily be combined to drive the PDM module. If the sine wave acted as the carrier and the other wave acted as the modulating signal that entered the scaler then amplitude modulation should be possible. If the carrier wave was in the RF range then the wireless transmission of this data may also be possible. If the frequency of the carrier wave were chosen carefully then the signal should appear in the AM broadcast band. The actual verification that an RF signal that was detectable by a standard AM receiver, though, was not determined until this phase of the laboratory was prototyped.

## 3 Assignment Overview

The scope of the overall assignment was summarized in the following manner: The specific DSP algorithm that is to be modeled in this laboratory is Pulse Density Modulation, PDM, which allows a one-bit digital output pin to mimic the functionality of a true analog output. Students will be asked to verify operation of the core Verilog HDL module by observing the waveform that is produced using an oscilloscope and digital multimeter. They will then modify the original module to playback prerecorded audio through a speaker that is connected to a digital port on the Terasic DE2-115 trainer. They will then be asked to reconfigure the module so that it can use pre-recorded audio to amplitude modulate a 1 Mhz sine wave allowing the audio to be received using a standard AM broadcast band receiver.

### 3.1 Pulse Density Modulation

Pulse density modulation is a common DSP technique that can be employed to create high-fidelity audio output in portable hand held devices. This form of modulation generates a binary bit stream, *b[0], b[1], ... b[n]*, where *b[i]={0,1}*, that causes digital voltage pulse values to occur in a sequence that when averaged corresponds closely to the targeted voltage of the analog signal that is being approximated. In Pulse Density Modulation in a positive logic two-voltage digital system, for each step in the binary stream, the output voltage can be represented as

$$V_{out}(i) = V_{max} \quad \text{when } b[i] = 1 \text{ and} \qquad (1)$$
$$V_{out}(i) = V_{min} \quad \text{when } b[i] = 0\,.$$

For the digital outputs on the Altera FPGA of the Terasic DE2-115 being used in this lab this corresponds to $V_{max} = 3.3\ V$ and $V_{min} = 0\ V$. If the pulse frequency of the binary bit stream is sufficiently larger than the frequency of the desired analog signal, then connecting the digital output to a simple low-pass filter makes it possible to produce any voltage within the $V_{max}$ to $V_{min}$ range.

Encoding the binary bit stream of the targeted analog signal using PDM utilizes the process of delta-sigma modulation. In PDM the analog signal is in effect passed through a one-bit quantizer that produces an output bit of b[i]=1 or a b[i]=0 in the bit stream in a manner that incorporates a mathematical representation of the actual digital voltage output and a pending quantization error. This quantization error is computed from the difference from the actual digital output voltage level associated with b[i] and the desired analog voltage level. This accumulated error is then negatively fed back in the process loop where it either adds or subtracts from the cumulative error which will in turn affect the threshold value that is used to quantize the next bit that is produced. PDM has the effect of causing the average error to be distributed out across the bit stream which results in the analog voltage being closely approximated at any point in time by the bit stream average.

Figure 1 illustrates the basic pulse density modulation algorithm implemented as a C/C++ function. There are two inputs. One input is the reference signal that is assumed to be within the range -0.5 to +0.5. The other input is a scale value which is always between 0 to 1. The actual signal to which the PDM method is being applied is the product of the original signal and this scale value times 2 which will be in the range of -1 to 1.

The C/C++ implementation shown in Figure 1 of this function utilizes 32-bit floating point mathematics while the Verilog version shown in Figure 2 utilizes 32-bit integer mathematics where the values have been scaled to reflect a fixed-point representation. Both give about the same precision for the data ranges associated with this application.

```
int pdm(float x,float scale) {
    int ret_val;
    static float ge=0;

    x=x*scale*2;

    if (x>=ge) {
       ret_val = 1
       ge  += 1-x;
    }
    else {
       ret_val = 0;
       ge  += -(1+x);
    }
    return ret_val;
}
```

**Inputs:**
**x = desired value of waveform at the sample point.**
**scale = amplitude scale factor**
**Output:**
**ret_val = binary output {0,1} which has an average that approximates this desired waveform value**

**Figure 1: Pulse Width Modulation Algorithm written in C++**

Figure 3 illustrates the fixed-point interpretation of the signed input variables, ***x***, and ***scale***, and the internal variable ***x_total***. Because of its **signed** data type designation, the most significant bit of each of these variables is the sign bit and the number is represented in its twos-complement form. Declaring these input variable as **signed** means that comparisons such as greater than (>) and algebraic operators such as multiplication (*) will be performed in a manner where the most significant bit will be interpreted as the sign bit. Since the value of the quantity ***x*scale**2* should always be between -1 and 1, the alpha sigma method employes both the addition or subtraction of the value 1 (as shown in Figure 1), which means that the value of the global error, **ge**, can at a given point in time be either positive or negative. This is the reason for using the **signed** data type. Fixed point values have the benefit that circuitry to implement arithmetic functions is easier to create and the amount of logic resources needed is much less that for a given floating point representation. The speed of fixed point operations is also significantly faster and much more predictable.

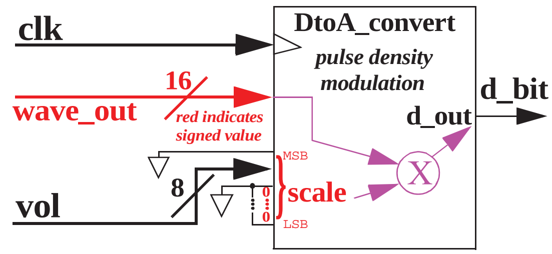
It is at this point were students are shown how the fixed point format can easily be emulated using signed integers. Fixed-point assumes that there are a fixed number of binary places to the left and the right of a binary point. In fixed point the integer (mantisa) portion of the number is to the left of this point and the fractional component is to the right. In standard signed integers this binary point is always assumed to occur after the left-most least significant bit position in the number meaning that there is no fraction portion of the number only an integer portion. As in many cases with low-level number representation, the value of the number depends upon the rules one uses for its interpretation. As long as we do not interfere with the function sign bit it is possible to assume a different weighting of the numbers that represent the integers by assuming that the binary point is associated with any bit position. This is shown in Figure 3 for the number **x** where bit positions 13 through 0 are considered to be to the right of the binary point and bit positions 14 and 15 are to the left of this point. Because the **signed** type is used bit 15 will represent the sign bit. This assumed weighting makes it possible to represent numbers in the range of -2 to 2 as shown in the figure. Verilog is unaware of this alternative interpretation we are imposing. To obtain the actual number one would divide the integer number by the factor of 2 raised to the number of bit positions to the right of the binary point (or shift right that many bit positions). Thus the number 1.0 for input ***x*** in Figure 2 could be represented in Verilog as 16'h4000, or as 16'd16384. (Using the same rationale the number 1.0 for ***x_total*** in Figure 3 is encoded as 32'h200000 in the verilog representation of the **pdm** task in Figure 2).

```
task pdm(output d_out,input signed [15:0] x,
   input signed [15:0] scale,
   inout signed [31:0] ge);
   // fixed point encoding of 1.0... assuming
   // signed 3 bit mantisa/29 bit fraction
  `define ONE 32'h20000000
   // 32 bit holder of x * scale data
   reg signed [31:0] x_total;
   reg ret_val;
   begin
   x_total = x*scale*2; // scaling value
   if (x_total >= ge)
      begin
      ret_val = 1;
      ge = ge + (`ONE - x_total);
      end
   else
      begin
      ret_val = 0;
      ge = ge - (`ONE + x_total);
      end
   d_out = ret_val;
   end
endtask
```

**Figure 2: Pulse Width Modulation Task written in Verilog**

The *pdm* task can be called from within a procedural *always* statement as shown in Figure 4. Note that this procedural block has been named *DtoA_convert*. The global error term ***ge*** needs to be declared outside the pdm **task** because it will be overwritten each time the block is executed.

The *DtoA_convert* element is active on the rising edge of the ***clk*** signal. During each rising edge it executes the *pdm* task creating a pulse in accordance with the PDM algorithm. It is driven by two signals. One is the ***vol*** signal which comes as an input to the module. The other, ***wave_out***, signal is an internal reg that is driven by the *Waveform_Gen2* procedural element.




```
always @ (posedge clk)
   begin:DtoA_convert
   // declaring local variable to
   // named section DtoA_convert
   reg signed [31:0] ge; // error term
   pdm(d_bit,wave_out,{1'b0,vol,7'd0},ge);
   end
```

**Figure 4: Procedural Implementation of PDM Sub-Component**

## 3.2 Phase I -- Waveform Generator Design

The overall functional design for the first phase of the laboratory assignment is shown in Figure 5. The student is responsible for designing the *Waveform_Gen2* module and implementing it in Verilog HDL. Its function is to periodically outputs data to the *DtoA_convert* section. The inputs to this module come from the DE2-115 board and include the system clock and a 12-bit frequency input. The ***clk*** frequency is assumed to be 50 Mhz and the output rate is assumed to be 50 times the frequency of the desired waveform. The value of ***freq*** was defined to represent the desired frequency of the waveform in Hz. The **vol** inputs to the *DtoA_convert* also are driven by signals from the DE2-115 board.

The experimental setup used in all phases of the design are shown in Figure 6. It requires the connection of the audio amplifier, multimeter and the oscilloscope to the digital port. In the phase of the assignment, students are asked to report on what they see and hear as they vary the volume and frequency of the sine wave generator they have created. They

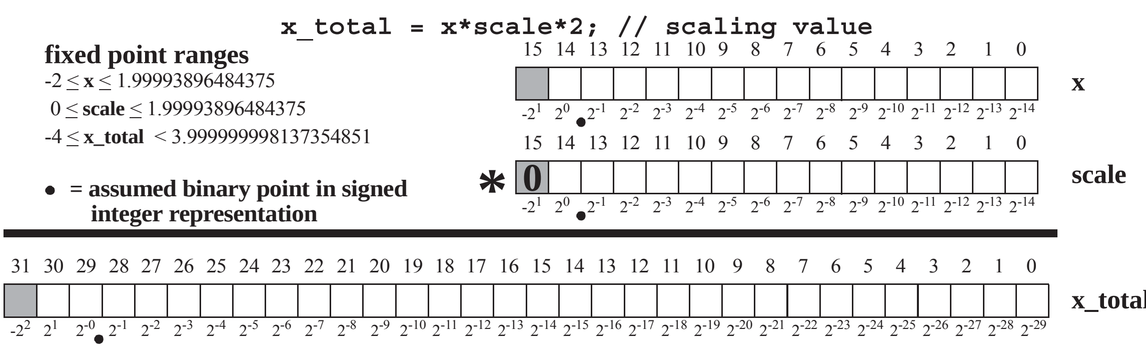



**Figure 3: Signed Fixed-Point Range Analysis**

are also asked various questions that relate to quantization effects, range, accuracy of the waveform that was generated.

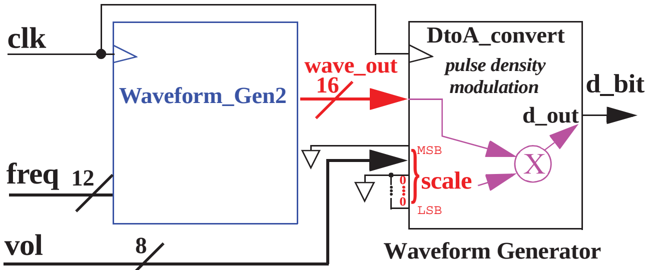



Figure 5: Overall Phase I Design Configuration

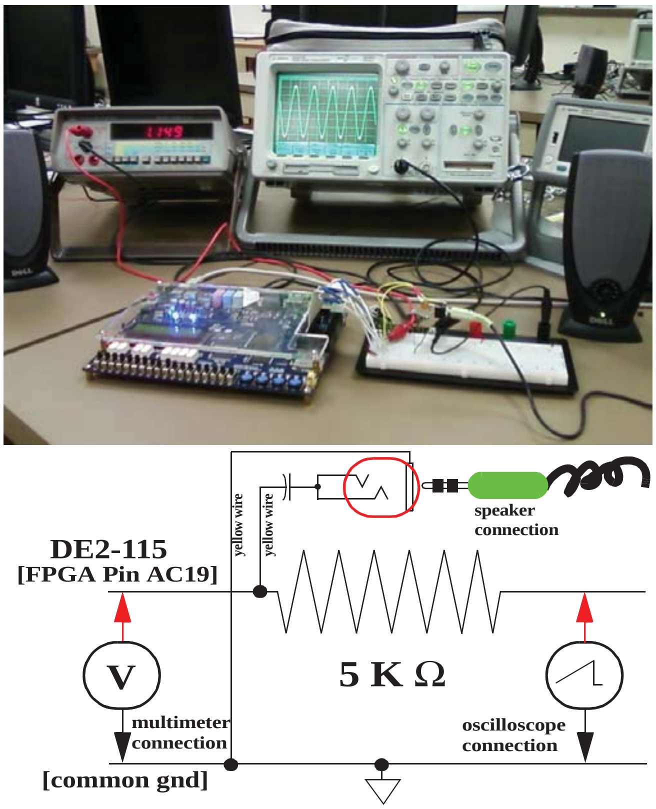



Figure 6: Common Experimental Setup

## 3.3 Phase 2 -- Audio Playback Design

In this part of the laboratory, students were asked to modify the *Waveform_Gen2* portion of the previous design so that it was able to output pre-recorded audio that had been previously sampled at 8000 Hz. This audio was placed in a set of text files that were obtained from various public domain sources. There were formatted in standard Verilog hexadecimal format. The data was encoded in two's complement form which was prescaled to allow for full volume to occur when the scale input is at its maximum value. Students were asked to employed two dimensional arrays and made sure that their model was encoded in a style that would allow the audio data to resided in the dedicated memory elements that were present within the FGPA. Figure 7 shows the general configuration for the overall Phase 2 design.

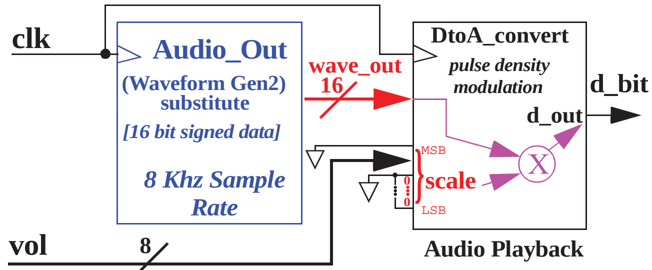



Figure 7: Overall Phase II Design Configuration

## 3.4 Phase III -- Wireless Audio Design

In the final portion of the laboratory, students were asked to combine the previous two parts to allow a 1 Mhz sampled sine wave, which is in the Radio Frequency, RF, range, to be amplitude modulated by the 8 Khz sampled audio. This allows the audio to be broadcast a short distance wirelessly where it was received by a standard AM receiver. Since the signal voltage levels are very low, and the effective transmission antenna is very small compared the 1 Mhz sine wave the RF emission levels will be well within the range allowed by the Federal Communication Commission, FCC, for unlicensed transmission.

Modulation is the process by which a RF signal can be made to transmit useful information. In this case this useful information is the audio. In amplitude modulation, as shown in Figure 8, the amplitude portion of the higher frequency carrier signal is made to change dynamically in proportion to the modulating signal, *M(t)* which is always non-negative (positive or 0). In the case where the modulation signal is symmetric around a positive and negative $V_{max}$ constant then an offset modulating waveform can be created by adding this $V_{max}$ constant to the two's complement representation. This is what has been done to the waveform files that are available to drive the audio in the design for this part of the laboratory. Once the audio waveform has been offset by this constant then amplitude modulation occurs when this sampled audio signal is multiplied with the sampled higher frequency sine wave as shown in the figure.

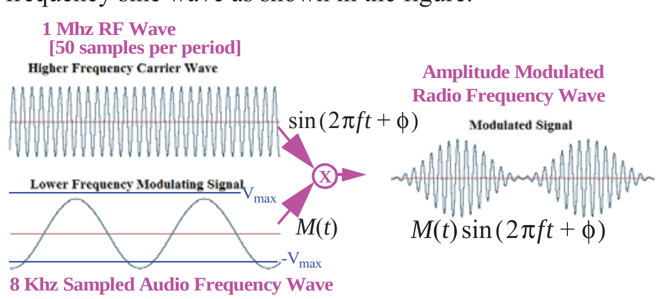



Figure 8: Basic Amplitude Modulation Process

Figure 9 illustrates the Phase III configuration. In this portion students must modify their *Wave_Gen2* module developed in the first phase of this assignment so that it outputs one of the 50 sample portions of the sine wave on each clock cycle thereby producing a 1 Mhz sine wave. They also had to modify their *Audio_Out* module so that it utilized data that is in the offset data format instead of the standard signed binary data format.

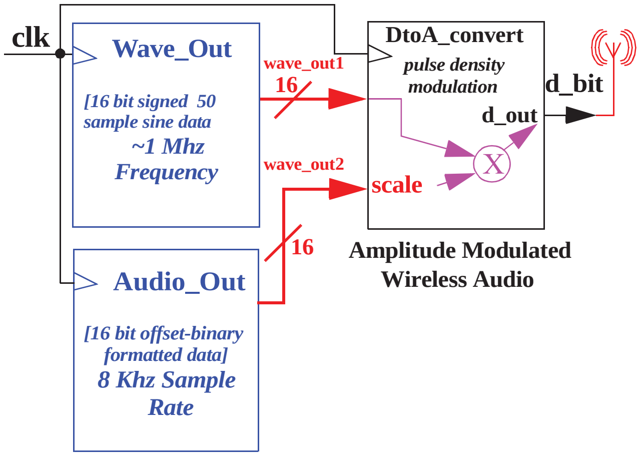



**Figure 9: Overall Phase III Design Configuration**

# 4 Initial Results and Conclusions

While the effectiveness of this assignment in promoting the learning of digital design concepts has not been measured quantitatively, anecdotal data seems to support that it was effective. Student and Graduate Teaching Assistant feedback for this Spring 2015 assignment was positive. The author also observed much ad-hoc experimentation that was occurring by the students in the laboratory during its initial phase. This was noteworthy because this was the portion of the assignment that was viewed by the author as being the least interesting. The ability of the students to dynamically change the frequency and volume, view the result on the oscilloscope, and multimeter, while listening to the audio waveform on the speaker seem to interest the students greatly. This may have been due to the fact that it simultaneously engaged the senses of touch, vision, and hearing. Something that today's students are very familiar with because of their experiences with portable electronic devices and various video gaming environments. While this lab experience would certainly not compare with such entertainment it was at least a mild reward for their diligence.

This laboratory assignment also allowed students to develop and expand their skills in such areas as electronic instrumentation (oscilloscope and multimeter) and hardware description language design capture and modelling. It exposed them to the application of a real-world DSP algorithm (PDM) as well as many DSP related issues. The assignment also required that the students become aware of the attributes and practical use of fixed point arithmetic. It also exposed them to how basic concepts from physics and AC circuit theory could be manipulated by their design to allow for audio and RF waveform generation.

This laboratory was also used to show students that it is possible to think-out-of-the-box so to speak and use elements in a manner that they were not originally design to be used. Using a digital port to produce an analog signal is one example of such a thought process. Another is driving this signal in at radio frequency levels in order to enable wireless communication.

The laboratory assignment also served to illustrate the various trade-offs that must be made. One trade-off was the size of the sampled audio data that used in Phases II and III. Students also had the option of using 8 bit data files. Using the smaller files reduced the fidelity but also reduced the on-chip FPGA storage, which resulted in less computationally intensive and less complicated place and route operations by the CAD tool. This drastically reduced FPGA synthesis time.

The final phase of the assignment also served to show that attributes of technology should be view neutrally, being neither good or bad. In a previous laboratory assignment students were shown that the electromagnetic effects of nearby conductors in high speed digital circuits could cause incorrect operation due to crosstalk. In this assignment it was shown that similar effects can be used to enable wireless audio transmission.

The bottom-up design paradigm worked well for this problem since the concept was simple enough to allow a basic understanding of each component before it was used. Obviously, this is not possible for all designs. The best approach is probably to incorporate both top-down and bottom-up structured assignments as well as at least one less structured more open-ended problem. This is likely to be the approach that will be taken in the future in this laboratory.

A major concern of many is that today's engineering students have grown up in a data rich environment which has favored the rote search for precomputed solutions to solve problems over the use of critical thinking skills. It is hoped that this and future laboratory assignments that will be developed will illustrate that while it is wise and prudent not to "recreate the wheel" a true engineer has the ability to take this "wheel" and improve upon it. The combination of design reuse and innovative ideas is what is likely to change tomorrow's technological landscape.

# 5 References

[1] M. Abdulwahed and Z. K. Nagy, "Applying Kolb's Experiential Learning Cycle for Laboratory Education," *Journal of Engineering Education*, 98: pp. 283–294. 2009

[2] S. M. Loo B. E. Wells, R. K. Gaede, "Exploring the Hardware/Software Continuum in a Computer Engineering Capstone Design Class using FPGA-based Programmable Logic," *Proceedings of the 2001 International Conference on Microelectronic Systems Education*, Las Vegas, NV, June 17-18, 2001.

[3] Rhonda Kay Gaede, B. Earl Wells, An Integrated Facility for Rapid Prototyping, *ASEE Annual Conference*, Saint Louis, Missouri, June 18, 2000.

[4] Wikipedia contributors. "Pulse-density modulation" Wikipedia, http://en.wikipedia.org/wiki/Pulse-density_modulation, Wikipedia, The Free Encyclopedia, 14 Jan. 2015. Web. 27 May. 2015.

# Video Games to Motivate a Multi-disciplinary Capstone Experience

**Daniela Rosca**[1] **and Jessica Kostiou**[1]
[1]Computer Science and Software Engineering Department, Monmouth University, West Long Branch, NJ, USA

***Abstract -*** *The Software Engineering Program offered by the Computer Science and Software Engineering Department at Monmouth University is one of the pioneer undergraduate programs in software engineering in the country. From its inception it required the students to participate in a sequence of two Practicum courses in order to develop a large-scale project, for an external client, working in teams. This year, one of the external clients, Invisible Wound, suggested a video game project for promoting the awareness of PTSD (Post Traumatic Stress Disorder) for children. This project idea has fostered the collaboration between the software engineering, art, and music students, in an attempt to incorporate multidisciplinarity in the software engineering Practicum courses. This paper summarizes the experience, lessons learned and plans for permanently offering this kind of projects for the senior software engineering students.*

**Keywords:** Capstone project, interdisciplinary, software engineering, video games design and development

## 1 Introduction

The Software Engineering Program offered by the Computer Science and Software Engineering Department at Monmouth University is one of the pioneer undergraduate programs in the country. From its inception it required the students to participate in a sequence of two Practicum courses in order to develop a large-scale project, for an external client, working in teams. Students have the freedom to choose one of the client suggested projects, or propose their own project ideas. Being an ABET accredited program, we have constantly provided opportunities for the students to work in interdisciplinary teams. So far, this ABET criterion has been fulfilled by working in group projects in classes outside the major, such as business, physics, or cross-cultural. This year, one of the external clients, Invisible Wound, a non-profit organization, suggested a video game project for promoting the awareness of PTSD (Post Traumatic Stress Disorder) for children. This project idea has fostered the collaboration between the software engineering, art, and music students, in an attempt to incorporate multidisciplinarity in the software engineering Practicum courses.

This paper summarizes the experience, lessons learned and plans for permanently offering this kind of projects for the senior software engineering students.

## 2 Practicum Course Sequence

Software Practicum is a sequence of two courses offered in the senior year to create the framework for students to put together the knowledge acquired throughout the program, in the context of a larger scale project developed for an external client, which can be a non-profit or for-profit organization. The projects vary in complexity, starting as simple as creating a website that requires backend support, and increasing in difficulty towards applications as complex as a course management system, or a workflow management system. The strength of the teams will dictate the type of project the team is willing to engage in.

Besides applying the software engineering principles and practices acquired in the program, students are also required to choose a software process model and use it throughout the development of the project. The final deliverables consist of the code and complete software documentation for each project. The documentation comprises a requirements document, architecture and design document, a test plan, a project plan, and user documentation. The students would have had the occasion to familiarize with all these types of formal documents during the specific classes on requirements, design, validation and verification, and project management. However, the Practicum project is the first time when they have to write all these documents for the same project, and realize the importance of creating consistent documentation that supports all phases of the project life-cycle.

The teams have three or four members that need to play roles of project manager, requirements manager, software architect, testing manager, and user documentation manager. The team members, following their strengths, assign these roles. The teams develop a project plan that encompasses the tasks necessary to create the end product, over two semesters. Also, the project manager is in charge of assigning weekly tasks, supervising their completion, communicating with the client, and with the class instructor.

The class is offered once a week, for two hours and 45 minutes. The instructor meets with each team to review the group's progress, the pieces of software documentation created by the team, the degree of completion of the tasks assigned for each team members, to learn of any problems the teams has, and try to find solutions for these problems. Following the team meeting, the team is working on the project for the rest of the class time. By the end of the class, the project manager creates an agenda for the work that needs to be done until the next class, and assigns specific tasks to each team member. This agenda will be used at the beginning of next class as support for the project status update. To foster collaboration among the team members, a common repository is created for all the project artifacts, using the tools such as: Github, SVN, Google Docs, etc.

One of the goals for this Practicum project is to test the students' life-long learning skills. Each project would require a technology that was not previously taught in any class. The students will have to teach themselves the new technology, demonstrating their readiness to do this for the rest of their professional careers. They are instructed to use the Fall semester to get familiar with all the technology needed for the project, such that, in Spring they can start the implementation phase.

During the development of the project, teams hold review meetings with the clients for clarifying the requirements specifications and the designs for their projects. At the end of the two semesters, the students need to formally present their projects in front of an audience composed of faculty, external clients, and peers. In order to properly prepare for this important event of the class, the teams are videotaped during a mock-up presentation, and together with the professor they analyze their performance, and are coached for a better performance during their formal presentations. The ABET teams that have looked at the videotapes of both mock-up presentations and final formal presentations, have noticed the significant difference in the quality of student performance.

The performance evaluation in this class is based on the quality of student deliverables, their timeliness, client feedback, peer evaluation, and instructor observation of student work throughout the two semesters.

# 3. The Battlemind Video Game Project

## 3.1. The game concept

Battlemind is an educational video game aimed to educate friends and family of war veterans of common behaviors associated with PTSD. The external client, Invisible Wound [1], is a charitable organization "which empowers warriors suffering from post-traumatic stress disorder to take an active role in their recovery and reintegration process." Their secondary mission consists of educating the public about issues surrounding post-traumatic stress disorder (PTSD). Another group of senior students has delivered a different Practicum project for this charitable organization a few years back. We are very pleased to see that they are trusting us again with another project that supports their initiatives.

Video-games development has been traditionally used to recruit more students and underrepresented minorities in the STEM disciplines, especially CS and SE programs [11]. Our intent is not to create an entire game development program [6, 11, 12, 13], rather allow students to experience this unique programming style and multi-disciplinary collaboration through the development of an educational video-game, as a senior capstone project.

The player of this game creates a schedule of typical daily events (see Figure 1), such as watching TV, taking the children to school, going to a movie theater, or sleeping through the night.

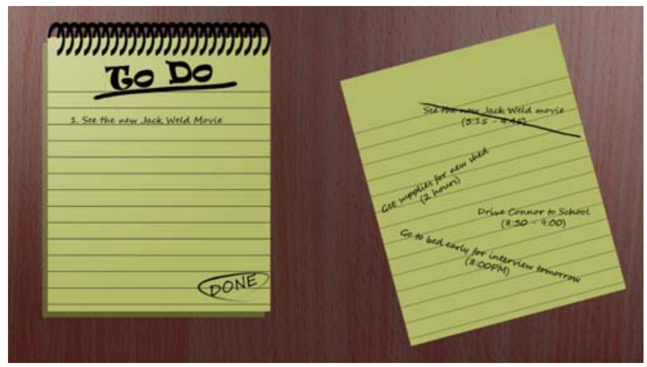



Figure 1. Creating the schedule of the day

The game can be interrupted by various mundane triggers, such as seeing a piece of garbage along the side of the road that reminds the warrior of integrated explosive devices (IEDs) or homemade bombs (see Figure 2), slamming of a door, watching war scenes on TV (see Figure 3), bumping into someone while finding a place in a dark, crowded movie theater (see Figure 4).

Figure 2. Taking the children to school, seeing objects along the road that can be possible triggers of PTSD symptoms.

Figure 3. Watching TV

When such triggers occur, the player needs to take an action to cope with them. The game will record the anxiety, fear, and depression associated with these events, and when a certain threshold level is reached, it suggests the usage of a set of cell phone applications to help cope with, and reduce the levels of these stressors. Examples of such applications are a breathing app, a messaging app, distress logger, event tracking, and eventually, calling of a friend (see Figure 5).

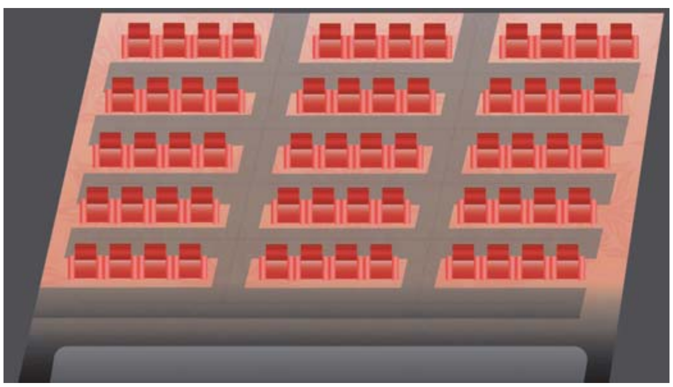


Figure 4. Finding a seat in a dark, crowded movie theater

Figure 5. Cell phone de-stress applications

## 3.2. Game design and development

The team in charge of developing this video game was composed of three software engineering students (will call them *engineers* from now on), one art student with a graphic design and animation background, and one student from the Music Department.

During the Fall semester, the engineers who were enrolled in the Senior Practicum course, researched the PTSD literature, held interview sessions with the client, and produced a game concept proposal that was reviewed with the client towards the end of the semester. The more engineering students were working on developing the game concept, the clearer it became that they would need professional help for the art and music assets. Having the game concept in place, the multi-disciplinary team was assembled at the beginning of the Spring semester. The graphic designer student and the music composer student were able to participate in our project by taking an Independent Study one-credit course, in their respective departments.

At the kick-off meeting all the students and the professors in charge of their supervision were brought up to speed regarding the background and goals of the video game. From that point on, the collaboration was done exclusively through the Google Docs repository, established for this project by the engineers. The engineers created a project plan, with milestones and deliverables necessary to implement the various scenarios of the game. The project plan was uploaded on Google Docs for the entire team. The project manager acted as liaison with the Art and Music members of the team.

The engineers took on the project manager, requirements manager, design architect, and testing manager roles. The Art student and the Music student were in charge of creating the graphical and sound game assets. The engineers chose Construct 2 [2], an HTML5-based game engine, designed for 2D games. Construct 2 has an intuitive drag-and-drop user interface for creating games with a very limited amount of coding required. Since the students had to learn the Construct 2 environment on their own, and they had the constraint of a short development time, this was a very appealing choice. Also, another advantage of Construct 2 is that the games created with this engine can be ported on a wide variety of platforms, such as internet-enabled Androids, iPhones, and PCs that recognize the CANVAS HTML element and associated JavaScript. The Art student used the Adobe Photoshop [3] and Illustrator [4].

The team chose an agile process model for the project development. Sprints were created for each game level. The team also created a set of minimal documentation to ensure proper internal communication. The documentation consisted of a requirements document, a game script together with an architecture document, and a set of test cases to be used to test the game implementation.

## 4. Lessons learned

The collaboration inside the multi-disciplinary team was challenging at times. Development of the music for the game went smoothly, with no issue. The composer was an honors student, very responsible and hard working. His deadlines were met every time, and the quality of the music delivered exceeded our expectations.

Unfortunately, we cannot state the same about the collaboration with the graphic designer. He was a senior having his own senior art show (the equivalent of the capstone project for the engineering students) happening in the middle of the Spring semester. Obviously, this was his highest priority. In the end, he delivered all the assets required of him, but the delay made it practically impossible for the engineers to finish all the planned levels of the game. Out of the initial five levels, only 3 were completed. This has prompted our attention to the lack of understanding between the members of the team from different disciplines of the challenges each of them had to face in this project: the engineers were seriously delayed by the late delivery of the art assets, while the graphic designer felt that he had not received substantial feedback from the engineers about his work. Time being the critical factor, when the engineers received an asset that was below their expectations, they preferred to keep silent and use their own graphical assets, thinking that there is no time to go back and redo a particular asset, when there were so many others that were still waiting to be completed. Having to drop features and levels from the game in order to complete it by the end of the semester is to be expected, taking into account that in the video-games industry, the development of a new game takes several years, employing teams much larger than ours.

Another issue that we identified at the end of the semester was the lack of face-to-face meetings of the multi-disciplinary team, after the initial kickoff meeting. Their collaboration was done exclusively through Google Docs. This was largely due to the unavailability of students to meet during the Practicum class, because of class scheduling conflicts.

It became apparent that instructors of the various disciplines need to communicate more and collaborate during the planning, which needs to take place before the Fall semester starts, and execution phases of the project. After this year's experience that has allowed the instructors to know each other well, the communication will be much easier for future projects.

## 5. Future plans

Based on this experience, on the feedback received from the students who have highly praised their participation in this multi-disciplinary project, and encouraged by the university's strategic plan, we have decided to create such opportunities for each senior class in the future. Learning from the shortcomings encountered this year, we plan on changing or adding a fresh approach to this multi-disciplinary endeavor:

1. Plan the type of educational game starting the summer before the Practicum project begins. Eventually, identify external clients or project champions from the School of Education or School of Nursing. Identify the team(s) of engineers who would like to work on a video-game project.

2. In the fall semester, ask the team of engineers to audit the Perspectives course "Critical Play: Theories and Aesthetics of Video Games" (PR425) and actively participate in the creation of the game concept, which will be the final project in PR425. The two courses will have to be scheduled during the same time frames. Also, ask the engineers to enroll in a Software Engineering Elective course that will teach Unity [5], the main game development engine on the market. This will be a newly developed course, that will be taught by an adjunct faculty with experience in a real laboratory for video-games development. It will allow the engineers to become comfortable with the game engine programming, and get ready to start the game development from the very beginning of the spring semester.

3. Ask the graphic designer(s) and music composer(s) to attend the final presentation of the game concept in PR425, which would help them become familiar with the game that will be developed next semester. Form the multi-disciplinary team(s).

4. In the spring semester, to facilitate the collaboration among the multi-disciplinary teams, we plan to offer the Independent Study courses in Art and Music at the same time with the Practicum course. This will allow the multi-disciplinary team to have face-to-face meetings every week. We believe that having to face your colleagues every week, will alleviate the number of missed deadlines, because the students will be able to see first hand the effect of their actions on the team members from other disciplines. We plan on asking the adjunct professor who teaches the Unity course in fall, to act as the production manager during the spring implementation of the game. He will be responsible for the coordination of the deliverables from various disciplines, keeping the team(s) on schedule, and assisting with eventual programming issues.

5. The critical interaction between the engineers and the graphic designers during the spring semester will take place according to the following plan:

| | |
|---|---|
| Week 1 | Create the game script based on the game concept finalized in PR425 |
| Week 2 | Make revisions and finalize the game script |
| Week 3 | Graphic designers create first level assets |
| Weeks 4-6 | Engineers implement Level 1 of the game; Graphic designers create Level 2 assets |
| Weeks 7-9 | Engineers implement Level 2 of the game; Graphic designers create Level 3 assets |
| Weeks 10-12 | Engineers implement Level 3 of the game |
| Week 13 | Test play and revisions |
| Week 14 | Final presentation of the game |

Table 1. Timetable for the development of the video game during the spring semester.

As can be determined from Table 1, we have decided that a reasonable game should not require students to develop more than three levels in a semester, to ensure a proper quality of the assets and the game.

## 6. Conclusions

This multi-disciplinary capstone project was challenging for both students and instructors, since there was no infrastructure in place for such a collaboration across disciplines, and also no planning for the orchestration of the project's execution. Our experience is similar to other attempts to use interdisciplinary teams for video-games projects [6-10]. However, through the lessons learned, we have come to acknowledge that the video-game project was also a worthwhile experience. The students have recognized that this was the most challenging, but rewarding experience of their undergraduate program. As one of them said: "It provided me with the experience in working on a large team and delegating tasks between many different areas of work. It gave me professional experience working with a client, and taught me the processes of how to create and present a product. Perhaps the most important lesson I learned is that things aren't always going to go as planned, and when that happens you need to be prepared."

It will probably take a few more video-games projects until we would have perfected the collaboration between students and instructors from the various disciplines. However, we are excited about this new curriculum idea, and are happy to share our enthusiasm with generations of future students who will be better prepared to enter a multi-disciplinary workforce.

## 7. Acknowledgments

The authors would like to thank professors Wobbe Koning, from the Art Department, and George Wurzbach, from the Music Department, for their contribution to this project, and are looking forward to a fruitful future collaboration.

## 8. References

[1] Invisible Wound. http://invisiblewound.org, 2015.

[2] Scirra. Construct 2. https://www.scirra.com/**construct2**

[3] Adobe. Photoshop. www.photoshop.com

[4] Adobe.Illustrator.www.**adobe**.com/products/**illustrator**.html

[5] Unity Personal Edition. https://**unity3d**.com/**unity**/**personal-edition**

[6] Hollie Bourdreaux, Jim Etheridge, Ashok Kumar. "Evolving Interdisciplinary Collaborative Groups in a Game Development Course"; Journal of Game Design and Development Education, Issue 1, pg. 25-37, 2011.

[7] Shannon Duval. "Creating a Games Class: A Walkthrough", ICFDG 2009, pg 279-284, ACM

[8] Quincy Brown, Frank Lee, Suzanne Alejandre. " Emphasizing Soft Skills and Team Development in an Educational Digital Game Design Course", ICFDG 2009, pg 240-247, ACM

[9] Bruce Maxim, Benjamin Ridgway. " Use of Interdisciplinary Teams in Game Development", ASEE/IEEE Ffrontiers in Education Conference, 2007, pgT2H-1- T2H-5

[10]Paul Gestwicki, Fu-Shing Sun, Benjamin Dean. " Teaching Game Design and Game programming Through Interdisciplianry Courses", Consortium for Computing Sciences in Colleges, 2008, pg 110-115.

[11] Jessica Bayliss and Kevin Bierre, "Game Design and Development Students: Who are They?", GDCSE'08, 2008, pg 6-10, ACM

[12] Joe Linhoff and Amber Settle, "Motivating and Evaluating Game Development Capstone Projects", ICFDG' 09, 2009, pg 121-128, ACM

[13]Robert Kessler, Mark van Langeveld, and Roger Altizer, "Entertainment Arts and Engineering (or How to Fast Track a New Interdisciplinary Program)", SIGCSE'09, 2009, pg 534-538, ACM

# The Need for An Integrated Effective Weather Dissemination System for Uganda

**Benjamin Kanagwa**[a,*], **Doreen Tuheirwe-Mukasa**[a] **and Khalid Muwembe**[b]
[a]College of Computing and Information Sciences, Makerere University, Kampala, Uganda.
[b]Uganda National Meteorological Authority, Kampala, Uganda

## Abstract

*Weather is one of the key factors that affect the livelihood of people in Uganda, in the various socio-economic sectors. Accessibility to reliable weather information is therefore vital for decision making. This paper presents the findings from a survey carried out to establish the status of the weather information dissemination system in Uganda. The survey sought to report on the current usage of weather information; inform on the current and future weather information needs for the different sectors; and guide appropriate modes for packaging weather information for different purposes; so as to forge a strategy that can improve the quality, access and impact of weather information. Results indicated a great interest in weather information; an expression by respondents for varying detail of weather information; and creation of personalized products. This provides the need for an effective weather dissemination system to provide more timely, accurate and accessible weather information.*

**Keywords:** weather information/forecast, effective weather dissemination, integrated system

## 1 Introduction

Most farmers in Africa including Uganda largely rely on the natural weather cycle for their crop and animal production. Indeed, many essential economic activities could also be better planned, and food security improved, if people were well informed of seasonal climate predications and could take appropriate actions [15]. Beyond the sparse meteorological network [10], Uganda faces a challenge of effective weather information dissemination to its diverse audience.

Weather specifically refers to conditions over a short period of time, the way the atmosphere is behaving, mainly with respect to its effects upon life and human activities. Weather consists of the short-term (minutes to months) changes in the atmosphere. Weather can change from minute-to-minute, hour-to-hour, day-to-day, and season-to-season [3]. Accessibility to reliable weather information is therefore vital for informed decision making. It leads to increased productivity (in the agricultural, energy, water resources and construction sectors) and safety (in the aviation, disaster management, fishing, health, mining, and defense sectors) [8]. However, communicating a forecast effectively requires not only sound science to support the information, but also a great deal of thought as to how, when, and by whom that information can and will be used in the context of daily life [9].

In Uganda, the effects of weather are pronounced in the socio-economic sectors of agriculture, aviation, construction, health, defense, disaster management and resource management. Considering the main livelihood of many people in Uganda arises from agriculture, which encompasses fishing, crop production and animal husbandry, weather information is more critical in the agriculture sector. Uganda's farmers and livestock herders have long been plagued by a lack of credible weather information, due to the authorities' inability to localize forecasts and package them appropriately without delays [11], yet farmers in developing countries have the potential to benefit significantly from weather and climate forecasts [13]. However, different stakeholders in agriculture will need weather information for different purposes. For example, a fisherman for one will be interested in the daily weather conditions that allow him to go out and catch fish, yet a crop farmer will be interested in seasonal weather information to know when they can plant, weed, harvest their crops while an animal keeper is interested in when they can spray their animals.
In the construction sector, individuals or companies target to sign a contract for a period of time of work when the weather is most favorable, and will least affect their activities, allowing for predictable delivery times. In health, availability of weather information is a central factor in planning for, mitigating or preventing outbreaks of diseases, while for disaster management will lead to reduction on the number deaths due to weather related conditions such as floods.

The Uganda National Meteorological Authority (UNMA) is mandated with promoting, monitoring weather and climate; providing weather forecasts and advisories to the Government of Uganda and other stakeholders for use in

*Correspondence: Benjamin Kanagwa, College of Computing and Information Sciences, Makerere University, P. O. Box 7062, Kampala, Uganda. Email bkanagwa@cis.mak.ac.ug

sustainable development of the country. In order to improve the quality, access and impact of weather information, the WIMEA-ICT (Improving Weather Information Management in East Africa for effective service provision through the application of suitable Information & Communication Technologies) project partnered with UNMA, as a key stakeholder of the research project to conduct a nationwide survey on Weather Information Dissemination (WID). The survey was carried out to establish the status of the weather information dissemination system in Uganda. The purpose of the survey was to generate and report on the current usage of weather information; inform on the current and future weather information needs for the different sectors and users; and guide appropriate modes for packaging weather information for different purposes. This was done to complement the efforts of the UNMA and related institutions towards meteorological data collection, management and dissemination.

The rest of the paper is structured as follows: Section 2 presents related work, section 3, the summary of the results of the survey and section 4 provides a motivation and rationale for the envisaged weather dissemination system. Section 5 presents the conclusion.

## 2 Related Work

In order for UNMA to achieve the mandate of providing weather information for all, one of the five factors that must be considered is an effective dissemination system for impact. According to UNMA, there are two main types of weather information that is currently disseminated. These are the short-range information, which spans from one hour to 5 (five) days, and the medium range information that spans up to 90 (ninety) days. This information is disseminated using different strategies. The short-range information is mainly disseminated by FM radio, email, television, or displayed on the UNMA[1] website. There are also tailored mobile weather alerts provided to fishermen (piloting in Kalangala district) and for farmers in Kasese district [15]. The information is also provided on the MTN[2] (one of the telephone service providers) website, which is a partner with UNMA in implementing the project. The principal language for the information is English. The same information is then translated in other local languages including Luganda, Lukonjo and Rutooro.
The medium range dissemination is for 10 (decadal) days, monthly and seasonal forecasts and is achieved through stakeholder workshops, where the stakeholders are selected from all the major socio-economic sectors; the National Media Centre, where a high profile person such as the minister of Water and Environment reads the weather statement before the media; local FM radios; and NGOs whose mandate is to facilitate the access of weather information by relevant stakeholder groups; on the UNMA website; through the Ministry of Agriculture early warning unit, and recently through the social media arena (via Facebook[3], Twitter[4], and U-tube[5]).
The information disseminated over the radio is simplified and translated into 20 languages/dialects to make it more understandable [2]. The seasonal forecast information is further accompanied by advisories on what actions the stakeholders can undertake, for instance what types of crops farmers can plant, relocating people from landslide prone areas to safer ones, installation of lightning arrestors on public buildings, strengthening general hygiene and sanitation, among others [6]. However, the dissemination done by UNMA is not without challenges, some of which are:

- Despite the fact that many of the messages have been simplified and translated, some stakeholder groups maintain that the messages received are still complex, as translation is done to the nearest approximate language
- A long standing attitude and bias by the general public that hinders acceptance of information relayed,
- Limited funding that only allows a few communication and dissemination channels to be explored.

These challenges are a subset of the six factors limiting the use of forecasts, as explained by [9]. They further posit credibility arising from the inaccuracy of past forecasts or the reputation of the messenger as one of the limiting constraints. Even though the media may generally be the most effective way of informing a large number of people, and at the lowest cost, it is unlikely to increase the trustworthiness of the forecasts. Additionally, cognition is another limiting factor. If users do not understand a forecast, they will use it incorrectly, or not at all. Generally, the limiting constraints all arise from poor communication practices [9].

As a preliminary to the survey, the research team interviewed officials from UNMA to discuss their stake in the project. As a result of this, it was established that UNMA envisions an interactive web portal where stakeholders can request for and obtain weather information in real time. They would also like to harness the social media capabilities to capture the interest, and encourage the participation of the youthful people. Furthermore, a Short Messaging Service (SMS) platform that allows individual inquiries to be made would be ideal.

According to [4], to fully exploit the benefits of meteorological forecasts, the forecasts must address a need that is real and perceived; be relevant; and provide decision options for stakeholders. Particularly, there is a need for appropriate interpretation of the forecasts; prediction of relevant components of weather for relevant periods, with sufficient accuracy and lead-time; presented in a form that can be applied if they are to impact decisions [13]. An effective

[1] www.unma.go.ug
[2] www.mtn.co.ug
[3] www.facebook.com
[4] www.twitter.com
[5] www.youtube.com

communication strategy should include efforts to help users understand the complicated information, preferably by giving them an opportunity to ask questions; include discussion of the local traditional indicators, and downscale the information to the local level, all of which can be achieved with a participatory system [9]. Moreover, broad distribution and operational use of forecasts beyond the life of a project must be supported by appropriate institutional dissemination channels, with safeguards to ensure quality, accessibility and timeliness of the information [5]. The need for a new service has become more pressing, as stakeholders say weather patterns are less predictable [11].

In view of this, there is need to understand the requirements and expectations of an integrated weather information dissemination system.

# 3 Results

The survey was achieved through a structured questionnaire that was administered to respondents who either completed the questionnaire with researcher(s) present, had an oral interview where the researcher(s) completed the questionnaire with the respondent's answers, or the respondents' filled out the questionnaire online. The questionnaire was pretested with a limited number of respondents to verify the clarity of the questions. Data from the questionnaires was analyzed using a spreadsheet program.

The survey was structured in 5 sub-sections. First, namely one, the general information that captured general aspects of the respondents who participated in the survey, including their district of location; the sector or their major economic activity; age; and gender; two, the channels of delivery that aimed at investigating how respondents get access to weather information; three, usage and accessibility of information, which sought the frequency of receiving; uses; usefulness of; impact; and interest in weather information by respondents; four, packaging of information which focused on presentation of weather information; and finally, five, issues and recommendations that encompassed suggestions of the respondents for effective dissemination. Table 1 presents an overview of the results of the survey.

Table 1: Overview of Survey Results

| **Socioeconomic characteristics** | | |
|---|---|---|
| Age | Frequency | % |
| 18-25 | 6 | 6 |
| 26-36 | 54 | 49 |
| 37-47 | 33 | 30 |
| >48 | 16 | 15 |
| **Sector coverage** | | |
| Resource Management | 24 | 22 |
| Agriculture | 21 | 19 |
| Aviation | 2 | 2 |
| Construction | 26 | 24 |
| Disaster Management | 16 | 14 |
| Health | 11 | 10 |
| Defense | 10 | 10 |
| **Gender** | | |
| Male | 26 | 76 |
| Female | 83 | 24 |
| **Currently receiving forecasts** | | |
| Yes | 75 | 67 |
| No | 34 | 30 |
| Not Aware | 3 | 3 |
| **Channels of Delivery** | | |
| Television | 64 | 36 |
| Radio | 41 | 23 |
| Newspapers | 20 | 11 |
| Physically from Meteorology Service | 6 | 3 |
| Email from Meteorology service | 14 | 8 |
| Meteorology Service website | 3 | 2 |
| Other websites | 10 | 6 |
| Mobile Phone | 13 | 7 |
| Other sources | 7 | 4 |
| **Efficiency of Weather Dissemination** | | |
| **Accuracy** | | |
| Very Bad | 8 | 7 |
| Bad | 18 | 17 |
| Fair | 6 | 6 |
| Good | 64 | 61 |
| Very Good | 9 | 9 |
| **Timeliness** | | |
| Very Bad | 8 | 8 |
| Bad | 27 | 27 |
| Fair | 7 | 7 |
| Good | 49 | 49 |
| Very Good | 9 | 9 |
| **Accessibility** | | |
| Very Bad | 9 | 8 |
| Bad | 26 | 26 |
| Fair | 8 | 10 |
| Good | 46 | 46 |
| Very Good | 10 | 10 |

## 3.1 General Information

The key sectors that consume weather information that were visited included Agriculture, Aviation, Construction, Defense, Health, Disaster and Resource management. The focus and purpose of the survey was to visit the selected sectors, and a total of twenty-one districts were visited all being representative of the sectors. The choice of district was taken from the existence of the sector in the district. It was also found that access to weather information generally reduces as the age increases. In terms of the gender, most of the respondents were male (76%), although the health sector had more female respondents than male.

## 3.2 Channels of Delivery

It was found that 30% of the respondents do not receive weather forecast information, 67% receive, and 3% were oblivious. Of the 67% who receive the information, most claimed to receive the weather forecast information at the end of a news bulletin either on TV or radio, as shown in Table 1. Some respondents received weather forecast information through newspapers, and others through various applications on mobile telephones. Table 3 shows a summary of the popular and easiest ways of access to weather information for the different sectors.

Table 3: Ways of Access for Sectors

| **Sector** | **Popular ways of access (in decreasing order)** |
|---|---|
| Agriculture | Radio, TV, email from meteorological service, experience |
| Aviation | Physically from meteorological service |
| Construction | TV, newspaper, radio, physically and email from meteorological service |
| Defense | TV, radio, newspaper, other websites |
| Disaster Management | Radio, Email from meteorological service, TV |
| Health | TV, mobile phones, radio |
| Resource Management | TV, newspaper, radio, mobile telephones |

Figure 1 presents the level of ease of understandability of the format and language used in the forecasts and severe weather warnings by the respondents.

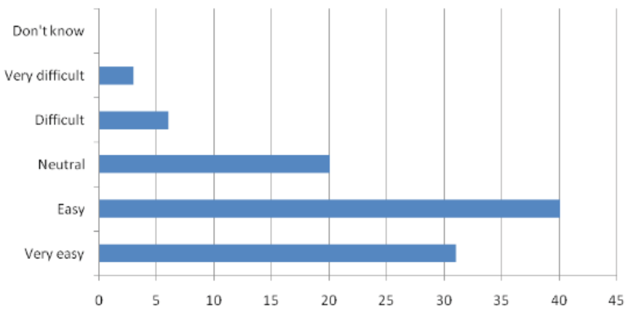



Figure 1: Understanding format and language of forecasts

## 3.3 Usage and accessibility

It was found that 86% of the respondents are affected by weather in their day-to-day activities, while 10% of the respondents think that they are not affected by the weather forecast information and 4% of the respondents did not respond to the question. The respondents who are affected use weather information for their plans and decisions, all sector-dependent. Respondents from construction were most affected, followed by those in resource management, agriculture, disaster management, with an equal number in defense and health. The respondents who claimed no effect were generally nearly equally spread across the sectors, with the health sector having the largest number. Figure 2 shows the frequency with which the respondents who receive weather information use it.

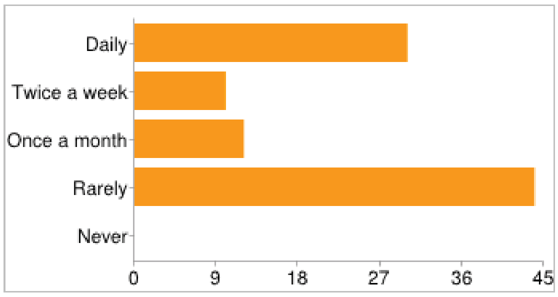



Figure 2: Frequency of use of weather forecast

The survey further sought the respondents' interest in weather forecast information. Most interest was expressed by the respondents from the sector of agriculture, followed by those in resource management, disaster management, and construction. Most respondents from the sector of health were somewhat interested in weather information. Several reasons were given for the levels of interest expressed, all related to sector driven activities. The respondents rated the usefulness of the weather forecasts and warnings of severe weather received, with most (48%) of them finding the information very useful as Figure 3 illustrates. Respondents mainly from resource management, construction, agriculture and disaster management rated them as very useful.

17% of respondents found the weather information somewhat useful and pointed out that often times the information provided is irregular, inaccurate and unreliable. These respondents were mainly from the health sector. Some of these therefore preferred to rely on the experience and knowledge of past seasons as passed on from their ancestors and observing the skies for changes so as to determine the upcoming seasons, especially for the sector of agriculture.

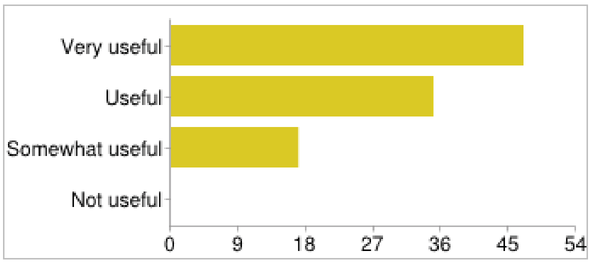



Figure 3: Usefulness of weather information

The survey also investigated aspects of weather concerning its accuracy, timeliness and access. By ranking of sectors, respondents from construction, resource management, disaster management, agriculture, defense and health generally found accuracy of the weather information good. In terms of timeliness, respondents from resource management and construction generally found the information timely, and the same sector respondents ranked access of weather information as generally being good. Even though about the same percentage of respondents found these aspects very good, and generally thought the information is timely, respondents from the sectors of agriculture and health generally found these aspects of weather information to be fair and bad respectively. Table 1 shows a summary of these aspects for the sectors. Figure 4 shows the frequency with which respondents would like to receive daily weather updates. 43% prefer to receive updates on a need to know basis (on demand), especially for the sectors of agriculture, resource management, and disaster management. 13% prefer to receive hourly updates, 10% after three hours, and 34% after six hours.

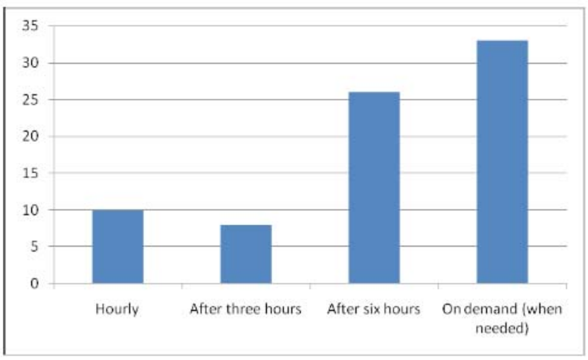



Figure 4: Frequency of daily Updates

On comparing the forecasts and warnings of severe weather provided today to two years ago, respondents stated that they are generally more useful and the information about forecasts themselves more available, in comparison to 5 years ago.

## 3.4 Packaging of Information

The respondents from the sectors of agriculture and health were interested in weather information for a season, while those from aviation for a day, specifically three hourly. The respondents from the sector of disaster management were equally interested in weather information of a day; three days in advance; and a month. In terms of location, these respondents from all the sectors were mostly interested in weather information for their regions, basing on the activities they undertake. They ranked the regions in terms of the country, district, village, county and parish. Table 4 shows the elements of interest to the respondents ranging from rainfall, temperature, wind, sunshine, hail and cloud cover.

The research team found that 49% of the respondents preferred to receive a detailed, but not technical weather forecast. They preferred a summary with the following aspects:

- Critical parameters highlighted, such as amount of rainfall received and expected; speed of wind and the direction; temperature of the day; among others
- Precise and short graphics,
- Tabular formats used to present the weather information
- Sent via SMS or other various phone applications
- Presented with captivating programs on TV, and well explained
- Packaged by regions or based on the users, e.g., for researchers and users from aviation sector, provide detailed information, whereas a summary may suffice for those from the agriculture sector
- Dedicated sections in the print media such as newspapers and magazines
- Be accompanied by advisories for the consumers to be able to take immediate or required action

## 3.5 Issues and Recommendations

Respondents advocated for assistance/training in the interpretation of the weather forecast. This was particularly for the sectors of disaster management, agriculture and health. More respondents in the sectors of aviation, construction, and resource management purported not to need as much training in the interpretation of the weather information. This is in tandem with the proposal made by [9] to circumvent the cognition constraint by not only simplifying the forecast, but teach users what the information means and how to interpret it.
Respondents suggested several ways to enable effective weather information dissemination including:

Table 4: Elements of interest in weather forecast

| | **Agriculture, Construction** | **Aviation** | **Defense, Disaster management** | **Health, Resource management** |
|---|---|---|---|---|
| Rainfall | † | † | † | † |
| Wind | † | | | |
| Temperature | † | | † | |
| Cloud cover | † | † | | |
| Sunshine | † | | | |
| Humidity | † | | | |
| Hail | † | | | |

- Disseminating the weather updates directly to people's phones as requested since majority of the people have mobile phones.
- Weather magazines specifying information of the previous month and projections in the new month to provide details of all weather parameters. These magazines could be delivered monthly
- Provide via Internet, or create an online all-time access Management Information System (MIS) capturing all regions of the country and the world available all the time.
- Simplifying terms used in the forecast dissemination presentations such as normal/below normal and others
- Create a data bank that can easily be accessed and that is regularly updated
- Use social media other than the traditional TV, Radio, and Newspapers channels.

## 4 Dissemination Strategies

In order to achieve effectiveness of forecast information, [12] argue that this depends strongly on the systems that distribute the information; the channels of distribution; recipients' modes of understanding and judgment about the information source; and the ways in which the information is presented.

Effective information dissemination requires a combination of pull and push technologies. Under the push technology, the user has no control on when the (or what) information is received, as they have not initiated a request for the information. These are commonly provided as broadcast messages. Examples of push technologies include the TV, Radio, newspapers and bulletins among others. On the other hand, pull technologies allow the user to decide on when to get the information. The information is obtained on demand whenever there is need.

One probable medium to achieve this is through the now ubiquitous social networks, a recent product of the ever-increasing advancement in online technology [16]. Many private companies are recognizing the potential uses of social network sites (SNS) as an excellent way to connect with their clients and receive their first-hand comments on what the companies or their products went well and what needs to be improved upon [14]. They additionally posit that social networks appear to be a very appealing form of communication among the younger users, including teenagers and younger adults. In their study to establish the viability of weather dissemination via social network sites, they found that the respondents to the survey that was carried out liked the idea of customizing the manner in which they received weather information. Furthermore, their respondents generally agreed to disseminating severe weather information via SNS, as the fastest way to be warned of impending severe weather, but were generally unsure of its credibility [14].

Mobile phone dissemination of weather information can also be explored, particularly the push and pull nature of SMS [7]. This is viable seeing as mobile telephony has increased dramatically in Sub-Saharan Africa over the last decade [1]. Mobile phones have also evolved from simple communication tools to service delivery platforms, which present several possibilities and opportunities. Local leaders in Uganda have advocated that mobile phones are a good medium for weather information dissemination, as long as there is credibility of the information people receive, and the information is received on time [11].

From the results of the survey to achieve efficient information dissemination, accuracy, timeliness and accessibility can be achieved through an integrated weather disseminations system that integrates both push and pull technologies. Integration also allows for the complementary efforts of different strategies to provide a fool-proof solution that pools together isolated efforts.

## 5 Conclusion

Access to timely and accurate weather information has continued to be a challenge for people whose livelihood is dependent on such information. However, this will remain a challenge unless an integrated weather information system is designed to address the diversity in information detail, timing, accuracy and accessibility of weather information by different stakeholders. The weather dissemination system needs to be robust, efficient and dynamic in the way it handles and manages weather information.

A multi-channel weather dissemination system is needed to deal with different audiences while disseminating the weather information system. The desired system must be accessible, anytime, anywhere. The possible system is an online, mobile platform for weather dissemination. The platform requires voice, textual, graphical and analytical weather information. The information provided by the system can also be used to create advisories for the different sectors.

# 6 Acknowledgement

We would like to acknowledge the WIMEA Research team for their contribution during different meetings and data collection.

# References

[1] J. C. Aker and I. M. Mbiti, "Mobile Phones and Economic Development in Africa" *The Journal of Economic Perspectives*, vol. 24, pp. 207-232, 2010

[2] D. Bamanya, The Climate Forecast Model, Uganda National Meteorological Authority, 06 Oct. 2014

[3] B. Dunbar. (2014) What's the Difference Between Weather and Climate? [Online]. Available http://www.nasa.gov/mission_pages/noaa-n/climate/climate_weather.html#.VLktXoWml2M.

[4] J. W. Hansen, "Applying seasonal climate prediction to agricultural production," *Agric. Syst.* 74 (3), pp. 305–307, 2002

[5] J. W. Hansen, "Realizing the potential benefits of climate prediction to agriculture: issues, approaches, challenges," *Agric. Syst.* 74 pp. 309–330, 2002.

[6] E. Kamuntu, September to December 2014 Seasonal Rainfall Outlook over Uganda, 2014.

[7] F. Mousumi and S. Jamil, "Push Pull Services Offering SMS Based m-Banking System in Context of Bangladesh," *International Arab Journal of e-Technology*, vol. 1, No. 3. pp 79-88, 2010

[8] J. Reuder and J. Sansa-Otim. 'WIMEA-ICT: Improving Weather Information Management in East Africa for effective service provision through the application of suitable ICTs', NORHED meeting, Bergen, 2013

[9] A. Patt and C. Gwata, "Effective seasonal climate forecast applications: examining constraints for subsistence farmers in Zimbabwe," *Global Environ. Change*, 12, pp. 185-195, 2002.

[10] J. S. Sansa-Otim and M. Waisswa, 'On Improvement of the Weather Information Management in Uganda', UBUNTUNET Connect-2012, Tanzania, 2012

[11] P. Sawa, 'Uganda takes stock of new climate information service', Thomson Reuters Foundation, 2014.

[12] P. C. Stern and W. E. Easterling, *Making Climate Forecasts Matter*. National Academy Press, 175, 1999.

[13] R. C. Stone and H. Meinke, "Weather, climate and farmers: an overview," *Meteorological Applications*, 13, pp 7-20, 2006.

[14] J.M. Wittrock et al., "Viability of Weather Dissemination via Social Network Technologies," presented at 90th American Meteorological Society Annual Meeting, Atlanta GA, 2010.

[15] WMO. (n.d) Reaching the Last Mile with Mobile Weather Alert The user-interface platform in action Volume 61(1) 2012 [Online]. Available: http://www.wmo.int/pages/publications/-bulletinarchive/archive/61_1_en/61_1_alert_en.html

[16] F. Zabini et al., "Communication and interpretation of regional weather forecasts: a survey of the Italian public," *Meteorol. Appl.*, 2014. DOI: 10.1002/met.1480

# Computer Science: Not about Computers, Not Science

**Kurt D. Krebsbach**
Department of Mathematics and Computer Science, Lawrence University, Appleton, Wisconsin 54911

**Abstract—** *This paper makes two claims about the fundamental nature of computer science. In particular, I claim that—despite its name—the field of computer science is neither the study of computers, nor is it science in the ordinary sense of the word. While there are technical exceptions to both claims, the nature, purpose, and ultimately the crucial contributions of the beautiful discipline of computer science is still widely misunderstood. Consequently, a clearer and more consistent understanding of its essential nature would have an important impact on the awareness of students interested in computing, and would communicate a more informed perspective of computer science both within academia and in the larger society.*

**Keywords:** computer, science, education, algorithms

## 1. Not about computers

Pick up almost any book on the history of computer science and Chapter 1 will be devoted to the evolution of the machines [1]: Pascal's adding machine, the brilliant designs of Babbage's Difference Engine and Analytical Engine, Turing's Bombe and Flowers' Colossus at Bletchley Park, and the ENIAC at the University of Pennsylvania, to name a few. It is no secret that every computer scientist is fascinated by—and enormously indebted to—these and many other magnificent achievements of computing machinery.

But is our discipline called "computer science" (CS) because we use the scientific method to empirically study these physical computing machines? I believe that few computer scientists would answer affirmatively, for it is not a computing machine that is the chief object of our study: it is the *algorithm*.

### 1.1 The centrality of algorithms

I am certainly not the first to claim that CS has been around long before modern electronic computers were invented; or in fact, before *any* computing machines were invented.[1] Let us instead consider that CS began as a discipline when an algorithm (i.e., a procedure for achieving a goal) was first discovered, expressed, or analyzed. Of course, we can't know precisely when this was, but there is general agreement that Euclid (mid-4th century BCE) was among the first computer scientists, and that his method for computing the greatest common divisor ($GCD$) of any two positive integers is regarded as the first documented algorithm. As Donald Knuth—author of the discipline's definitive multi-volume series of texts on algorithms—states: "We might call Euclid's method the granddaddy of all algorithms, because it is the oldest nontrivial algorithm that has survived to the present day" [3]. Knuth's version of Euclid's $GCD$ algorithm [4] is shown in Figure 1.

| | | |
|---|---|---|
| E1. | [Find remainder.] | Divide $m$ by $n$ and let $r$ be the remainder. (We will have $0 \leq r < n$.) |
| E2. | [Is it zero?] | If $r = 0$ the algorithm terminates; $n$ is the answer. |
| E3. | [Interchange.] | Set $m \leftarrow n, n \leftarrow r$, and go back to step E1. |

Fig. 1: Euclid's 2300-year-old algorithm for finding the greatest common divisor of two positive integers $m$ and $n$.

### 1.2 Requirements of algorithms

Let us use this simple example to state the requirements of a valid algorithm (adapted from [5]):

1) It consists of a *well-ordered* collection of steps. From any point in the algorithm, there is exactly one step to do next (or zero when it halts).
2) The language of each step is *unambiguous*, and requires no further elaboration. The "computing agent" (CA) responsible for executing this algorithm (whether human or machine) must have exactly one meaning for each step. For example, in $GCD$, the CA must have a single definition of "remainder".
3) Each step is *effectively computable*, meaning that the CA can actually perform each step. For instance, the CA must be able to compute (via a separate algorithm) a remainder given two integer inputs.
4) The algorithm produces a *result* (here, an integer).
5) It *halts* in a finite amount of time. Intuitively, each time E1 is executed, $r$ strictly decreases, and will eventually reach $r = 0$, causing it to halt at E2.

Finally, a mathematical *proof of correctness* exists to show that $GCD$ is *correct* for all legal inputs $m$ and $n$, although correctness itself is not a strict requirement for algorithms. (As educators we see more incorrect, but still legitimate, algorithms than correct ones!) The importance of an algorithm's close relationship to mathematical proof will become apparent shortly.

[1] Indeed, the term "computer" was originally used to refer to a person. The first known written reference dates from 1613 and meant "one who computes: a person performing mathematical calculations" [2]. As recently as the start of the Cold War, the term was used as an official job designation in both the military and the private sector.

## 1.3 Algorithms as abstract objects

Note that so far we have assumed no particular computing device. No arguments about PCs vs Macs, and therefore debates over the merits of Windows, OSx, and Linux are moot. In fact, we have not even been forced to choose a particular programming language to express an algorithm.

So algorithms—our fundamental objects of inquiry—do not depend on hardware. Our interest is not so much in specific *computers*, but in the nature of *computing*, which asks a different, deeper question: What is it possible to compute? For this, Alan Turing provides the answer.

### 1.3.1 Turing machines

Alan Turing, rightfully considered the father of modern computing, provided us with this insight before the invention of the first general purpose programmable computer. In 1936, Turing presented a thought experiment in which he described a theoretical "automated machine" that we now call a *Turing machine*. A Turing machine (TM) is almost comically simple, consisting of an infinitely-long tape of cells upon which a single read/write head can scan left and right one cell at a time. At each step, the head can read a symbol, write a symbol, move left or right one cell, and change state. An example TM instruction is shown in Figure 2. Turing further showed that a TM, given instructions

| | |
|---|---|
| IF | state = 3 and symbol = 0 |
| THEN | write 1, set state to 0, move right |

Fig. 2: A sample Turing machine instruction.

no more complicated than in Figure 2, can calculate anything that is computable, regardless of its complexity. Several corollaries of this result provide insight into the nature of the abstract objects we call algorithms:

- A TM can simulate any procedure that any mechanistic device (natural or artificial) can carry out.
- A TM is at least as powerful as any analog, serial or parallel digital computer. They are equivalent in power, and can implement the same set of algorithms.
- Any general-purpose programming language that includes a jump instruction (to transfer control) is sufficient to express any algorithm (Turing-equivalence).
- Not every problem has an algorithmic solution.

So anything a physical computer can compute, a TM can too. The most advanced program in the world can be simulated by a simple TM - a machine that exists only in the abstract realm...no computer required! This is an important reason why CS is not fundamentally about physical computers, but is rather based on the abstract algorithms and abstract machines (or humans) to execute them.[2]

[2]However, see Section 1.5 for more on the obvious *usefulness* of implementing and executing programs on modern computers!

## 1.4 Evaluating algorithms

Computer science is not only about *discovering* algorithms. Much research is also devoted to *evaluating algorithms*. Here are some questions we ask when we compare two algorithms that achieve the same goal:

1) **Correctness:** Is the result guaranteed to be a solution?
2) **Completeness:** Does it always find a solution if one exists?
3) **Optimality:** Is the first solution found always a least-cost (best) solution?
4) **Time Complexity:** How many units of work (e.g., comparisons) are required to find a solution?
5) **Space Complexity:** How much memory is needed to find a solution?

A core area of CS involves evaluating algorithms independently of computer, operating system, or implemented program. The first three questions pertain to the type of answer an algorithm yields. Note that none of the answers to these questions depend on particular computer hardware or software.

Questions 4 and 5 pertain to an algorithm's *computational complexity*, which measures the amount of some resource (often time or space) required by a particular algorithm to compute its result. An algorithm is more *efficient* if it can do the same job as another algorithm with fewer resources.

But there is a counterintuitive insight lurking here. Even concerning issues of efficiency, it is of little use to know how many resources are required of a specific *implementation* of that algorithm (i.e., *program*), written in a specific *version* of a specific *programming language*, running under a specific *operating system*, on specific computer *hardware*, possibly hosting other processes on a specific *network*. Too many specifics! Instead, we want our hard-earned analysis to transcend all of these soon-to-be-obsolete specifics. Although space constraints preclude a sufficient explanation here, the sub-discipline of computational complexity provides a well-defined way to classify algorithms into classes according to the "order of magnitude" of the resources required as a function of the input size of the problem, and provides another illustration of the centrality and abstract nature of algorithms.

## 1.5 Implementing and executing algorithms

One of the most important and profound differences between modern CS (as a branch of mathematics) and pure mathematics is the ability to write computer programs to *implement* algorithms and then *execute* these programs on different sets of inputs without modifying the program.

This ability—to actually *automate* computation rather than simply describe it—is by far the most practical benefit of modern computing. For this reason, it is also the most visible aspect to the general public, and even, I would claim, to other academic disciplines (who tend to equate "computer scientists" and "coders"). But although everyone agrees on the important role of programming, programs and algorithms are not equivalent. Abelson and Sussman express this distinction eloquently in their classic text: "First, we want to establish the idea that a computer language is not just a way of getting a computer to perform operations but rather that it is a novel formal medium for expressing ideas about methodology" [6]. In other words, programming is important not only for the machine-executable code produced but also to provide an appropriate set of abstractions with which to contemplate and express algorithms.

## 2. Not science

The authoritative Oxford English Dictionary defines CS as "the branch of knowledge concerned with the construction, programming, operation, and use of computers." This paper takes the opposite view. How can a "science" be *primarily* about construction, operation, or use of any machine?

Fellows and Parberry argue that: "Computer Science is no more about computers than astronomy is about telescopes, biology is about microscopes or chemistry is about beakers and test tubes" [7]. And while I have always liked the accuracy and clarity of the quotation, it is true because neither science *nor mathematics* is about tools. Computers are tools, and the general-purpose computer is about as useful as a tool can get. Engineering is about tools, and computer engineering is about designing and building incredibly useful tools for CS and really for every other discipline as well.

### 2.1 Mathematics, science, and engineering

Many would acknowledge that what we mean by the term "computer science" is an odd blend of mathematics, science, and engineering. Paul Graham even argues for the decomposition of the field into its component parts when he admits: "I've never liked the term 'computer science'... [it] is a grab bag of tenuously related areas thrown together by an accident of history, like Yugoslavia" [8]. Similarly, John McCarthy, coiner of the term "artificial intelligence" and inventor of the second-oldest programming language (LISP), implored practitioners to keep the distinction clear: "Science is concerned with finding out about phenomena, and engineering is concerned with making useful artifacts. While science and engineering are closer together in computer science than in other fields, the distinction is important" [9].

### 2.2 Procedural epistemology

In emphasizing the central role of *computational process* and *abstraction* in CS, Abelson and Sussman state: "Underlying our approach to this subject is our conviction that 'computer science' is not a science and that its significance has little to do with computers" [6]. They go on to associate CS with mathematics in describing the computer revolution as "the emergence of what might best be called *procedural epistemology*", but distinguish the two by saying, "Mathematics provides a framework for dealing precisely with notions of 'what is.' Computation provides a framework for dealing precisely with notions of 'how to"' [6].

I think the brilliance of this statement lies in the term *procedural epistemology*, i.e., that fundamentally we are in search of which procedures (algorithms) *exist*, and which can not exist (as Turing and others have proven). One of the central claims of this paper is that CS is fundamentally a branch of mathematics - but not because the most important algorithms compute results that are of interest to mathematicians. It is because *algorithms themselves* are abstract objects in the same way that *proofs* are abstract objects. They are not physical, but nonetheless exist, waiting to be discovered.

### 2.3 Empiricism

The primary definition of "science" from Princeton's Wordnet is: "The study of the physical and natural world using theoretical models and data from experiments or observation" [10].

While I have already discussed the first part, I would argue that the second part of the definition involving the empirical aspect of science is likewise not the chief method of investigation in CS (although there are recent exceptions, as discussed in Section 3.2). Science follows the scientific method, which involves forming falsifiable hypotheses, devising experiments to test them, and observing the results with the hope of building converging evidence for or against them. This model of investigation is appropriate for observing and predicting natural (and evolving) processes, but not for eternal objects such as proofs and algorithms, which are not directly observable. Surely, as our knowledge grows, we develop a richer set of abstractions with which to express and investigate them, but the objects themselves have always been there and always will be, waiting to be "discovered" by those interested in the "how to."

Ultimately, because algorithms are abstract and exist in neither the natural (or even physical) world, and because we therefore do not use the scientific method as the primary tool to investigate them, CS is not fundamentally science.

## 3. Caveats

My intent in this paper is to confine my claims to what is "primary" or "fundamental" to the discipline of CS, noting in several places when claims should not be interpreted as absolute or exclusionary. I now briefly discuss several of these important exceptions.

### 3.1 Computer hardware

While the design and manufacture of computer hardware is more closely identified with electrical engineering, students of CS can benefit greatly from instruction in hardware design, especially in the interface between the native instruction set of the computer (software), and the various levels of hardware implementation of those instructions; in fact, the best computer scientists are those who can cross over to related disciplines and see the same objects from multiple perspectives. Learning to think like an electrical engineer makes one a better computer scientist, and *vice versa*, but that does not make one a subset of the other.

### 3.2 Empirical science

While I have argued that computer scientists do not primarily employ the scientific method, the relatively new sub-discipline of *experimental algorithmics* is a promising exception to this rule. In her recent text, Catherine McGeoch explains: "Computational experiments on algorithms can supplement theoretical analysis by showing what algorithms, implementations, and speed-up methods work best for specific machines or problems" [11]. While the idea of empirically measuring the performance of implemented programs has long been in the toolbox of practicing software engineers, the idea of a *principled* study of strategies for tuning and combinatorially testing algorithms and data structures is newer, and promises to be of great practical value. However, even McGeoch explicitly suggests that experimental algorithmics fill a *supplemental* need in the discipline, and that the student can only benefit from these new methods by first having a solid working knowledge of the principles of algorithm analysis and data structures.

## 4. Societal impact

Despite serving as an umbrella term for a group of variously related efforts involving mathematics, science, and engineering, the central questions of CS revolve around abstract objects we call *algorithms*: identifying them, discovering them, evaluating them, and often implementing and executing them on physical computers. Thoughtfully unpacking the confusion between computers and computing can have important benefits for how the larger society—including future computer scientists—perceives of and ultimately considers our useful and fascinating discipline.

### 4.1 Public perception

At one end of the public spectrum are those who see computer scientists as *super users*; humans who have formed a special bond with computers (if not other humans?) and are therefore capable of inferring the correct sequence of menus, settings, and other bewildering incantations without ever having read the manual. Fortunately, as society has become more technically savvy this misunderstanding is fading, and perhaps the majority instead think that computer scientist is another name for *super programmer*.

Of course, this misconception is less problematic, as programming *is*, in fact, an important aspect of CS, as described in Section 1.5. And, in fact, many computer scientists *are* expert programmers. But, useful as they are, programs are not the core of CS. Algorithms exist regardless of whether someone has coded them, but not *vice versa*. Algorithms come first. There are excellent computer scientists whose research does not involve much programming (e.g., analysis of algorithms), just as there are excellent programmers who might not do much interesting CS beyond coding to detailed specs. But the conflation of these terms can lead people to mistakenly believe that an undergraduate curriculum in CS should consist of writing one big program after another, or learning a new programming language in every new course. These are simply not accurate reflections of the work computer scientists actually do.

### 4.2 Undergraduate expectations

My experience as an academic adviser has helped to motivate this paper. With the proliferation of mobile computing and omnipresent Internet, there *appears* to be a lot of enthusiasm for our field, but I often find that eager undergraduates are either excited about something that is *not* CS (gaming comes to mind), or stems from an advanced topic that definitely *is* CS (e.g., machine learning), but requires a strong foundation in algorithms, data structures, and programming abstractions to place the advanced concepts in context. Understanding the state of the art gives promising students the best opportunity to contribute something genuinely new.

Unfortunately, students have been led to believe that there are massive shortcuts to these fundamentals…online tutorials, cut-and-paste code, developer boot camps, and so on. And while each of these methods can have their uses, they are no substitute for the foundation that will serve a student throughout an entire career. In rare cases, shortcuts result in short-term success, but fundamental abstractions provide the foundation upon which to continue building and adapting throughout a lifetime of learning and innovating.

## References

[1] P. E. Ceruzzi, *Computing: A Concise History*. The MIT Press, 2012.
[2] D. A. Grier, *When Computers Were Human*. Princeton, NJ, USA: Princeton University Press, 2007.
[3] D. E. Knuth, *The Art of Computer Programming, Volume 2 (3rd Ed.): Seminumerical Algorithms*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1997.
[4] ——, *The Art of Computer Programming, Volume 1 (3rd Ed.): Fundamental Algorithms*. Redwood City, CA, USA: Addison Wesley Longman Publishing Co., Inc., 1997.
[5] G. M. Schneider, J. Gersting, and S. Baase, *Invitation to Computer Science: Java Version*, 1st ed. Pacific Grove, CA, USA: Brooks/Cole Publishing Co., 2000.
[6] H. Abelson, G. Sussman, and J. Sussman, *Structure and Interpretation of Computer Programs*, 2nd ed. New York, NY, USA: McGraw-Hill, Inc., 1997.
[7] M. R. Fellows and I. Parberry, "SIGACT trying to get children excited about CS," *Computing Research News*, vol. 5, no. 1, p. 7, Jan. 1993.
[8] P. Graham, *Hackers and Painters: Essays on the Art of Programming*. Sebastopol, CA, USA: O'Reilly & Associates, Inc., 2004.
[9] J. McCarthy, "Merging CS and CE disciplines is not a good idea," *Computing Research News*, vol. 5, no. 1, pp. 2–3, Jan. 1993.
[10] Princeton University. (2010) About wordnet. [Online]. Available: http://wordnet.princeton.edu
[11] C. C. McGeoch, *A Guide to Experimental Algorithmics*, 1st ed. New York, NY, USA: Cambridge University Press, 2012.