Curriculum Design, Development and Assessment for Computer Science and Similar Disciplines

I. Schagaev¹, E. Bacon², N. Folic³ and N. Ioannides⁴,*

¹Faculty of Computing, London Metropolitan University, 166-220 Holloway Road, London, N7 8DB, UK.
²School of Computing and Mathematical Sciences, University of Greenwich, Maritime Greenwich Campus, Old Royal Naval College, Park Row, Greenwich, London, SE10 9LS, UK.

Abstract - Curriculum design and development for Computer Science and similar disciplines as a formal model is introduced and analysed. Functions of education process as knowledge delivery and assessment are analysed. Structural formation of curriculum design is presented using definitive, characteristic and predictive functions. The process of change in the discipline is described and analysed. The algorithm to determine the core of the discipline is developed. Functions of the core moving and merging are introduced and analysed. A new assessment technique, multiple choice answers approach (MCAA) with several right and not mutually exclusive answers from a larger set of answers, is introduced. MCAA, even though developed as an assessment technique, also facilitates the formation of reliable coverage of subject curricula. The rigorous formation of assessment becomes an essential segment of curriculum design and development. Levels of applicability are explained showing the practical effectiveness of the proposed approach. A trial application in a real module of computer science showed promising results. Schemes of student progress assessment are introduced including analysis of success and protection from guessing, using penalty functions. Schemes of further development and directions for further research are discussed.

Keywords: Curriculum Design, Curriculum Development, Curriculum Core, Theory of Classification, Formalisation, Multiple Choice Answer Approach, Penalty Function.

1 Introduction

Computer Science is a relatively new and fast growing discipline for teaching. It absorbs different theoretical and technical results from different disciplines and creates a fusion which penetrates and influences many aspects of human life. Computer Science is, in fact, the theoretical base for the fastest ever growing area of technological development, that of Information Technology (IT). At first glance, Computer Science as a discipline should absorb its own technical applications (IT), but previous attempts to do this failed. It is, therefore, necessary to create a ‘bridge over troubled water’ and make curriculum design, development and assessment more connected and, where possible, rigorously designed.

The need for such an effort evidently grows day by day, as one of the fathers of Computer Science, Dijkstra, in his letter [1] to the Communications of the ACM, admits: “I would therefore like to posit that computing’s central challenge, viz. "How not to make a mess of it," has not been met. ... You see, while we all know that unmastered complexity is at the root of the misery, we do not know what degree of simplicity can be obtained, nor to what extent the intrinsic complexity of the whole design has to show up in the interfaces. ... To put it bluntly, we simply do not know yet what we should be talking about...”

This work attempts to form a logical core of curriculum design for computer science and similar disciplines. Initially, the main terms used are defined and the role of education as a very important driving force in the improvement of life for the society at large is identified. Existing models of curriculum development and their drawbacks are briefly discussed. Curriculum design as information processing is then analysed and further developed by the introduction of three main functions in discipline construction: definitive, characteristic and predictive. The core of the discipline, the way of its selection and its main features: moving and merging, are analysed and discussed. The authors also focus on the issue of assessment and address this as part of curriculum development and a new type of assessment, Multiple Choice Answers Approach (MCAA) is introduced, and is supported by a scheme for its implementation as well as suggestions for a penalty function. Our goal is to create a new assessment methodology, which, apart from assessing the students, also enables us to estimate the quality of teaching delivery. The creation of an automatic (or semi-automatic) assessment procedure along with the development of an algorithm for the selection of the most important material for assessment will help in estimating the quality of teaching delivery. A result from such an assessment for a concrete module from Computer Science is presented. Finally, further work and open problems in the further development of our approach are outlined.

By considering the following facts about Computer Science and other similar disciplines, one realizes that the need for this work becomes multi-fold:

- Computer science in general suffers from a snowball of information, useful and otherwise [1], which cannot be handled, properly processed or justified for teaching purposes.
• Standard analysis of the learning outcomes mostly serves to indicate the general performance through the use of statistics and has very limited module or student meaning.
• The level of competency of students varies enormously.
• Even though social demand to help students with different backgrounds get high levels of education is present, the resources to match these demands are limited.

If the strategic aim of this work is to be achieved, Computer Science as a discipline for teaching must be modified to incorporate IT, tuned for purpose, thus resulting in a more efficient and effective teaching process (those who ruined us, help us).

2 Science, knowledge, skills, curriculum: definitions and classifications

The main terms used in this study are general and have a variety of meanings which depend on the human activity that they may be used for. It is, therefore, necessary to define the specific meaning of these terms for the work presented here. Complete description of each term can be found in [2].
• Science - 1. The study, description, experimental investigation and theoretical explanation of the nature and behaviour of phenomena in the physical and natural world; 2. Branch of systematized knowledge of study.
• Knowledge - 1. Information, understanding acquired through learning or experience; 2. The total body of known facts or those associated with a particular subject; 3. Justified or verifiable belief, as distinct from opinion (Phil).
• Skills - Special abilities in particular field acquired by learning or practice.
• Curriculum - The courses offered by an educational institution or followed by an individual or group; Latin - running, course, course of study, programme.
• Computer Science - Study of the construction, operation, and use of computers.

A more holistic approach to the word curriculum assumes that it should be placed between the aim of education and the learning outcome, where the aim is “what we want to achieve” and the learning outcome is “what we are able to measure”. A major question raised here about the learning outcome, as this assumes to express the result of education in one sentence, is whether we can actually do this!? This term will not be used here and its applicability is out of the scope of this work.

Curriculum design was analysed by Aristotle: “For the formal nature is of greater importance than the material nature” [3] and Confucius: “He who learns but does not think is lost; He who thinks but does not learn is in great danger” [4], clearly identifying the necessity of reflecting on what one has learned. This work aims to build an algorithm of Curriculum design and development for Computer Science and similar disciplines using our own recent theoretical results and experience during the redevelopment of an existing module within the Faculty of Computing of the London Metropolitan University.

The terminology shows that Science differs from knowledge by the indirect introduction of the Subject (an agent) to receive knowledge. In spite of its well-deserved recognition, the Penguin Dictionary has, in our opinion, a serious mistake in the definition of the word knowledge by using the too general and absolutely not essential in this context term information. Long discussions on the interrelations of this term are presented in the number of books written by N. Wirth, E. Dijkstra, W. Turski and others. Our opinion about the relation between knowledge and information is shown below and the efficiency of this description is further elaborated below:

Knowledge = Information + Algorithm of its Application  \( (1) \)

Curriculum, in turn, is just another name for the program of study. Just like any other program it must be complete, efficient and reliable in order to provide education in the selected area or discipline. The education cycle is completed when its outcome has been returned back to the society (figure 1). The roles of the main agents (student and teacher), are analysed here in the framework of the education process.

![Algorithm of Education to Society](image)

As shown in Figure 1, feedback is the main measure of teaching success. During teaching the lecturer is the main deliverer of the products: knowledge and skills. The main consumer or customer is the student. The cycle of education is completed when a customer returns his/her results back to the society and, therefore, becomes a deliverer.

Confucius indirectly confirms the usefulness of this kind of modelling by presenting a measure of success for education by the production of better men after education: “a goal is to create gentlemen who carry themselves with grace, speak correctly, and demonstrate the integrity of things”. The model
presented here goes a step further with the teacher becoming the main manufacturer and deliverer of knowledge, while the student changes their role from consumer during the learning process to a deliverer after it, when s/he returns back to society new experiences from the results of education, while Confucius claims that teacher is “a transmitter and not maker”.

Any discipline of Computer Science assumes a certain amount of knowledge and skills in some proportion which plays a very important role in discipline design, and therefore, curriculum development. Skills are different from knowledge in the definitions presented above as they concern a special ability, not a wide understanding of the area. A good argument in favour of this differentiation is the variation in the area of application – skills are concerned with the application of particular knowledge or experience in a specific, well known environment, whereas knowledge is about the application of experience and learning outcomes in an uncertain and wide environment. This separation is important in further discipline structure and its curriculum development.

The equation $K + S = \text{Successful Application}$ presents Knowledge ($K$) and Skills ($S$) as essential components of education. Rigorous separation of $K$ and $S$ helps to balance discipline structure by means of using the most appropriate instruments of the teaching/learning circle: lectures, tutorials, practicals, courseworks, as well as various assessment techniques. Three main but different types of segments (modules) of Computer Science as discipline are presented in Figure 2:

![Fig. 2. Ratio of knowledge and skills in computer science disciplines; the blue (darker) segment presents knowledge.](image)

Case A refers to areas within the discipline of Computer Science such as Algorithm Design and Problem Solving, Software Engineering, Network Technologies, Theory of Programming, Discrete Math, Theory of Data Bases, etc. Case B refers to areas such as Human Computer Interaction, Computer Aided Design, Computer Graphics, etc. Case C refers to areas such as Web Design, HTML, Java, Visual Basic, Modula-2, Pascal, Applications of Data Bases, etc.

In disciplines where $K > S$ the discipline must be built with a wide area of knowledge, be more abstract and, where possible, general in order to show the limits of the existing knowledge, as well as its place in the context of science. Areas of discipline which fall within this scenario require much more active lectures and seminars in the pure academic meaning of these words.

In disciplines where $K < S$ there should be more concern for deeper and practical aspects of one narrow area of skills and the applicability of these skills. Areas of discipline which fall within this scenario require many more practical sessions with small introductions of elementary or essential theory.

There is no doubt that the curriculum for these different cases should have different structure and forms.

The invention of measurement and efficiency of teaching and progress of learning by the use of the so-called Learning Outcomes (LO) caused both serious criticism and scepticism amongst practitioners. Indeed, when considering that 11 from 15 modules in Computer Science have nearly identical LOs and with all others being pretty much similar the immediate question which arises is: what do these LOs actually mean? Neither the structure of the discipline, nor its context and success of understanding can be described in two or three sentences.

It is clear from the above that new research is required in the area of structuring of knowledge delivery and derivation of schemes in order to measure the result of discipline delivery. The research work presented here focuses on schemes of knowledge delivery and assessing and in this context Curriculum Development.

### 3 Modelling of educational program development

Three very complex entities are involved in the process of education: Science, Science Deliverer (Lecturer) and Science Consumer (Student). Various authors [5][6][7] described the main elements of the course design process, their interrelation in time and their logical order. The most cited author [6] presents the generalized model shown in Figure 3.

![Fig. 3. A typical model of the course design process](image)

The model shown in Figure 3 suffers from poor logic: student characteristics are placed before content determination; goals and objectives, which are much more general terms and wider in meaning, are placed after determination of the context; teaching and assessment methods are selected based on goals and objectives; implementation and evaluation requires “adjustment” as necessary, but with no indication as to the meaning of the phrase “as necessary”. More logical might be the sequence presented in Figure 4.

![Fig. 4. Modified model of course design process](image)

But even this “logicalisation” (figure 4) cannot change the comments on the applicability, usefulness and efficiency
and their measurement in the course design process. Both models shown in Figures 3 and 4 are too general to be useful. Furthermore, the resource dependence of the discipline and possible ways of its delivery are not even mentioned in the models of these authors [5][6][7].

We must also add that the context of the discipline should be involved, embedded and reflected in and during the curriculum design process. These arguments make obvious the need to develop a new model of the discipline and, therefore, curriculum design. The order is important: at first, one has to develop a discipline, and then the way of delivery. Below is such a sequence of discipline development:

1) Determination and definitions of the main elements in the discipline:
   - Description of connections between definitions.
   - Selection of main features of definitions.
2) Analysis of the schemes for discipline delivery (either existing, or new).
3) Specification of discipline delivery main elements:
   - Course structure.
   - Assessment instruments.
   - Performance issues and scheme of its measurement.

Work published by [8][9][10] concern a development of the Theory of Information Processing, described at the level of the main categories and resources. Here, the input information goes through an algorithm (processing), new information is created on the way and the result is produced and delivered. The process of Learning can also be analysed as Information Processing and vice versa: The Processes of Teaching and Learning are in fact one entity – Information Processing.

4 Curriculum Development as Information Processing

By accepting this point of view we could apply terms and results from the theory of information processing for various aspects and areas of pedagogy – in this case to the processes of discipline and curriculum development. This means that principles of classification for information systems can help build a course of learning with the highest possible understanding as well as a measure of this understanding.

The next important issue relates to the analysis of all possible resources necessary to form a discipline and its curriculum. Rigorous classification of resources and ways of their use and processing should be built to form the framework for all further steps.

The delivery of knowledge is a process of transforming a student from a state without knowledge to another state, where the knowledge obtained can be assessed and confirmed. The process of assessment can differentiate those students with the required level of knowledge and those without. A well-known sequence of steps to deliver knowledge includes:

1) Knowledge delivery.
2) Delivery of practical (application of knowledge).
3) An assessment.

All steps in this sequence should be completed at the right time and in such a manner that the system (University) should notice neither a fault nor the process of its elimination. Students are involved in all these steps. Some may not understand some elements and/or may not be able to “process” information when delivered in the course, but can catch up during the course. The passing of an assessment will then confirm a student as successful in acquiring the required level of knowledge. In some cases, however, expected failure, lack of confidence or change of interest may cause a student to change his/her course. Such cases will not be considered here due to the uncertainty and existing elements of subjectivity that such cases entail.

In reality, many more steps (in the Knowledge Delivery Algorithm) are needed to eliminate malfunctions and avoid student failures or even their withdrawal from a course. These actions depend on the various functions and features of learning and the teaching processes, their roles and “power” to provide knowledge for the students.

A Knowledge Delivery System can be analyzed as an information processing system using first order philosophical categories such as matter and time. For Knowledge Delivery it is structure and time. More importantly, the main function of knowledge (information) delivery must be considered at the first level of description, together with matter (structure) and time (see figure 5).

Fig. 5. Classification of resources to deliver knowledge

The levels shown in Figure 5 describe the structural core of the system to deliver knowledge. While first order categories present reflections of our understanding (they exist in the mind only), the second order descriptions (hardware, software) present real world objects as executive elements for the first order categories.

In general, the above mentioned sequence of steps towards delivering knowledge should generally be completed in scheduled time and in such a manner that the ‘average’ student could cope with the speed of its delivery as well as be able to continue learning. Furthermore, attention should be paid to the following sensible comments:

1) Time spent on exams and tests, in fact, is excluded from learning.
2) This sequence does not match IT use.
3) There is no direct evidence that lectures delivered were transformed into knowledge until the assessment was completed and passed.

The first statement suggests that the time of learning and the time of assessment (tests, exams) are separated for the student and is usually wrong to expect that under pressure of a possible failure the student continues to learn during the assessment.

The second statement suggests that knowledge delivery uses IT in one-way, and assessment uses IT in another. Knowledge delivery primarily concerns presentation aspects, while the use of IT in assessment is often based on multiple choice questions.

The third statement suggests that even the best developed and presented lectures can be unclear for students with different background. It becomes visible only during exams, and by then it is too late.

What options and resources do we have when creating a discipline and curriculum suitable for efficient teaching and learning? Again, according to [8][9][10] these are structure, time and information. At the same time, it is clear that testing time (assessment) should be eliminated from the algorithm of knowledge delivery (see first statement above) even though without assessment it will be impossible to receive feedback on the quality of education, either at the macro level – from the society at large (see Fig.1), or from the university. To resolve this issue well-known hardware design techniques of self-checking developed by [11] could help and should be used. In our case the most important feature of self-checking is concurrency, or near-concurrency, with functioning and self-checking even though the term self-checking in pedagogy has a one-to-one synonym: self-assessment. The difference lies in the quality of self-assessment which is very weak in comparison with self-checking as described by the Boolean logic function. As far as resources for knowledge delivery are concerned time cannot be counted, as it is beyond our control! Figure 5 can therefore be transformed as shown in Figure 6.

![Fig. 6. Classification of resources to deliver knowledge excluding time](image)

The two blue (shaded) boxes above present Information Technology elements, which can be involved in the process of knowledge delivery. We believe that further progress in the theory of Knowledge Delivery should be found in the selection of the principles and also in finding ways to load and reload it on the IT resources.

5 Principles of curriculum design

By direct analogy with [8][9][10] the Knowledge Delivery System must realize three closely connected functions: Definitive, Characteristic and Predictive.

A Knowledge Area has to have a Definitive Function (DF), in which terms and concepts are called and nominated. DF answers the question “What is it?”. The second function describes the interrelations between Definitions and characterises them and is called the Characteristic Function (CF). CF answers the question “how are these definitions connected?” . The Predictive Function (PF), in turn, answers the question “What if?”. Knowledge of DF and CF and their application are essential elements here. Application of definitions and characterisation of their interrelations enables the use of elements that have just been learned and predicts their behaviour.

But what is the core for the successful organisation of a learning process for one discipline and what is it for another? According to [8][9][10] the success of learning is the formation of a strong PF in the selected area of Knowledge discipline and holds true for both new and well established disciplines. The success of PF depends on concurrent observation and satisfaction of two conditions:

- The first condition is accuracy, precision in the selection of the aim pursued, with analysis of a new or existing subject domain and the key required feature - kind of aim integrity.
- The second condition, not less important, is the internal structure of the course: rigorous approach in the construction and introduction of terms and concepts inside the course, as well as its assessment procedures.

To summarise, the requirements on the formation of a new curriculum note:

1) Any discipline, as an introduction of a theory, should be considered from the point of view of performance of three interconnected functions: Definitive, Characteristic and Predictive.

2) The discipline must be constructed with the strict principle of aim integrity, i.e. with the selection of the single feature (predictive function) necessary to achieve and maximise.

3) Each phase of introduction, presentation and detailed analysis of a discipline development must be rigorously analysed. Otherwise, the success of the course built around this discipline will be problematic.

4) Only really essential features and details of the analyzed discipline objects and phenomena must be included. For example, the course of computer science as information processing system should be discussed in terms based or directly connected with information; a course describing distributed systems and networks should present in some way categories of dimensions directly connected to these kinds of systems, as well as basic relative terms.

The applicability of the three functions mentioned above is the key to successful discipline and curriculum development. Essentially, the resources available should be analyzed, including any new technologies such as information.
processing. IT involvement in the process of course construction should be considered from the beginning. Here, we try to analyse the process of knowledge delivery from the point of view of three interrelated functions: DF, CF, PF and algorithm of learning, using the approach from [8][9][10].

Consider the sequence of functions discussed above:

$$\text{DF} \rightarrow \text{CF} \rightarrow \text{PF}$$  \hspace{1cm} (2)

At first one delivers definitions (DF) and then their interrelations (CF). The algorithm of learning completes when a learner has acquired the skill in the subject to predict the behaviour of the elements presented in the course, and PF exists. The power of prediction (PF) means understanding of the course material by the student. The bigger an individual PF is the better the student knowledge is. PF should be formally measured. Growth of the knowledge can be described as evolution and transition from nowhere to PF:

$$\text{Nowhere and Nothing} \rightarrow \text{DF} \rightarrow \text{CF} \rightarrow \text{PF}$$  \hspace{1cm} (3)

This evolution can be analysed and terminated when $\text{PF}_s > \text{PF}_t$ (indices $s$: student and $t$: threshold). Threshold indicates the required level of understanding, proven by assessment through analysis of predictive functions developed in the student’s mind.

If a full set of definitions exists in a discipline it can count towards some development of DF. The question how to build the DF of the subject has its own importance and requires special research and elements essential for this paper will be introduced further.

Suppose for now that a teacher has prepared the DF of the course, or its part, and is ready to deliver it (Nowhere and Nothing $\rightarrow$ DF). The first iteration of learning will then consist of the delivery of DF from teacher to student. Once the student absorbs the essential definitions required for the understanding of the lecture material information from the course segment DF is completed and s/he is ready to learn further. Connections of elements – terms in the subject denote (determinate) $\text{CF} \rightarrow \text{DF} \rightarrow \text{CF}$.

In math CF is an essential set of formulae, in language course – grammar, etc. If we can deliver to a student an explanation on how terms of the course are connected then we can assume that CF does exist. Then the process of learning goes to another phase:

$$\text{DF} \rightarrow \text{CF} \rightarrow \text{PF}$$  \hspace{1cm} (4)

In a discipline, as a whole, it is impossible to expect that DF has to be learned as a whole before a student starts to understand CF also as a whole. Therefore, there must be a way to break DF and CF down properly into logical pieces.

To build well-balanced “golden” shape segments of DF one has to build as correct sequence of several DF in a curriculum as a whole:

$$\text{DF} = \{\text{DF}_1, \ldots, \text{DF}_i, \ldots, \text{DF}_n, \text{DF}_u\}, \quad i = 1, n$$  \hspace{1cm} (5)

During the learning process a sequence of $\text{DF}_i$ for $i = [1,n]$ should be accompanied by a proper sequence of $\{\text{CF}_i\}$ with some shift in time, as presented in Figure 7. Any overlap in time with the delivery of $\text{DF}_i$ and $\text{CF}_i$ is acceptable and even useful. The length of overlap should be selected regarding the level and ability of students. Also, the size of a particular $\text{DF}_i$ might be determined by a student’s ability for learning. For good students any particular $\text{F}_i$ might be bigger than for weaker students. This is especially important due to some recent political initiatives such as that for ‘universities for all’.

$$\text{DF}_1 \cup \ldots \cup \text{DF}_i \cup \ldots \cup \text{DF}_m \cup \ldots \cup \text{DF}_n = \text{DF}_u$$  \hspace{1cm} (6)

The form of the curve for $\text{DF}_i$, $\text{CF}_i$ and $\text{PF}_i$ could also be different as well as the length of their intervals and would be defined by the features of the discipline and the personal preferences of the deliverer. It is not clear as yet what defines the form of curve and the length of the triangles except of the student and teacher abilities. It is, however, clear that the total workload of delivery of $\text{DF}_i$ is a sum of workloads to deliver all elements as shown in the equation below:

$$W(\text{DF}_i) = \int_{t_i}^{t_{i+1}} \frac{d\text{DF}_i}{dt} \quad \text{where} \; t_i \; \text{and} \; t_{i+1} \; \text{define the interval of delivery for} \; \text{DF}_i$$  \hspace{1cm} (7)

In a similar way, the total workloads to deliver CF and PF are shown below:

$$W(\text{CF}_i) = \int_{t_i}^{t_{i+1}} \frac{d\text{CF}_i}{dt} \quad \text{and} \quad W(\text{PF}_i) = \int_{t_i}^{t_{i+1}} \frac{d\text{PF}_i}{dt}$$  \hspace{1cm} (8)
Naturally, discipline delivery should be done smoothly, with well-balanced functions of $DF_1$, $CF_e$, and $PF_i$ as mentioned above. But proper management of their delivery seems to be a subject of special research, and includes human computer interaction aspects and specific features of the learning subject.

6 Curriculum Core

To build the core of a curriculum on a subject a set of Definitive Functions ($DF$) and one of Characteristic Functions ($CF$) must be constructed. The $[DF]$ should contain the set of terms, in the same way as the Glossary of a book does, and the $[CF]$ should contain the set of topics describing how definitions are connected, in the same way as the Index of a book does. By combining terms from the Glossary with the Index we have a semi-automatic procedure to form a set of essential questions required for Assessment. Up to now, except for pure math disciplines, the proper formation of a set of essential questions was hardly known. There is no doubt that statistics should be obtained to analyse the relevance of these questions to the core of the curriculum. Additionally, a logical sequence of the questions should be arranged for the assessment, as well as for the process of delivery of the discipline. It is clear that even a semi-automatic procedure might be quite useful, as it reduces teacher’s workload in formation of assessment and, at the same time, guarantees quality. In one extreme there is a fully automated option in formation of assessment by direct programming of a set of terms and a set of questions. But this approach requires further research and approbation. In another extreme, a “manual” application of this algorithm should be taken.

If the core of the subject does exist, it does not mean that questions and terms taken from one book or lecture notes do relate to the core. Personal preferences of a teacher can be so high that they may completely mislead students and should be avoided. Thus, several books on the same subject from different authors could be selected and a joint Glossary ($DF_j$) of terms, which includes terms that belong to at least one glossary from the analysed books and lecture notes, could be organised thus creating an essential Glossary ($DF_e$) of terms which includes those terms only common to all Glossaries. Formally these two sets of glossaries are described as follows:

$$DF_1 = DF_1 \cup DF_2 \ldots \cup DF_{n-1} \cup DF_n$$

$$DF_e = DF_1 \cap DF_2 \ldots \cap DF_{n-1} \cap DF_n$$

Building a set of questions around an essential glossary $DF_e$ forms an essential set of questions $CF_e$ and, in fact, complete as a whole a formal preparation of assessment. $DF_e$ and $CF_e$ combined form a core of the discipline assessment.

The core of the curriculum and its features present us with a special interest. Above we discussed in brief some elements of the algorithm how to define a core. Here we discuss features of the observation and behaviour of the core. We discover two phenomena in behaviour of the core – moving core and merging core.

A moving core occurs when curriculum design (and redesign, as this is a permanent process due to the appearance of new books, new papers and other sources of information) shows growth when using some descriptive elements ($DF_i$) and decrease of use for some others. This change of intensity in the use of different areas of the core is shown in Figure 8 by a different intensity of blue colour.

This process of movement can lead to the separation of a discipline into several new disciplines. This separation should be detected early before it happens, as the discipline can lose its predictive function ($PF$) and become obsolete – theory can become very advanced to be practical (pure math case) and vice versa – practice can be developed much faster than theory and thus theory becomes almost useless, having only a descriptive function (networking). In the first example, knowledge does not connect with skills and is therefore hardly applicable to the society at large; in the second, the technological development has big advances in comparison with theoretical adaptation of the technological results. Both cases are very inefficient in terms of the model about the role of education presented in Figure 1. Keeping in mind this model, education should be effective from the point of view of improvement to the quality of life, and thus the early detection of the process of moving core seems to be important.

On the other hand, the merging of several cores can be caused by serious inventions, practical demand or revolution in technology, which accelerate the involvement of different disciplines into fusion as shown in Figure 9 where Computer Science is used as an example. The main reason of this fusion is the growth of Predictive Function resulting in the applicability of a new discipline to society.
Such processes and their detection are pretty complex but understandable and their algorithmisation and programming do not look impossible. This is a perfect area for the use of IT to assist the teacher to assemble the correct elements for the core of the discipline.

In the long run, these two processes can help detect changes in scientific areas related to the discipline automatically and assist the teacher to almost automatically adjust the curriculum and assessment to these changes. There is hope here that artificial intelligence (AI) and IT can be involved much more in the formation, correction and adjustment of curriculum for any scientific or technical discipline.

7 Assessment

Methods with which an assessment using the model of curriculum design and development proposed [12][13] can be formed are discussed in this section. By considering the requirements for assessment, a nearly formal assessment procedure could be realized that is also free from existing drawbacks. This procedure is called Multiple Choice Answers Approach (MCAA) and should not be confused with Multiple Choice Questions (MCQ) as the two are very different. A process for the formation of a questionnaire and a scheme to maximize the efficiency of assessment, thus increasing the objectivity of the curriculum, are also introduced.

7.1 Assessment requirements

We consider the most important requirements and features of assessment to be:

- Objectivity.
- Quantitative analysis.
- Time (and other resources) efficiency.
- Concurrency with the learning process.

The popularity of assessing using MCQ, especially during the last decade, driven by the help that it provided teachers with the complex and boring procedure of marking and calculation of assessment results, has been growing very fast. And even though MCQ is a relatively new invention, which enables the applicability of IT to be used as a tool for assessment, it was considered a panacea. However, even though successful in that, the MCQ approach, disappointingly, does not provide proof of efficiency and growth of knowledge. The fact is that it does not have a connection with real delivery of knowledge or a measure of the quality of knowledge delivery – an assessment of its result. The standard assessment used to be limited by definitive descriptions, i.e. without the use of the set of functions, characteristic {CF} and predictive {PF}, as described in [9][10].

According to [14], Bertrand Russell claims that: “to be directly acquainted with something is to be in a position to give it a name in the strict logical sense, and to know something only by description is to know only that something uniquely fits the description”. But to give something a name and to be able to understand and use that name correctly are two different things …

7.2 Formulation of the Multiple Choice Answers Approach (MCAA) assessment

The Multiple Choice Answers Approach (MCAA) that enables the effective use of IT for assessment and matches the requirements set in section 7.1, is presented here. Compared with its predecessor, MCQ, the MCAA, which is based on the process of curriculum design and development presented in [12][13], enables the application of IT for assessment, increases the reliability of knowledge, eliminates guessing, and can help manage the assessment of thousands of students at the same time! Assume that the discipline in question, Computer Science or other similar, has sets of definitions (Definitive Function, DF), a set of key characteristics about how definitions are connected (Characteristic Function, CF) and, therefore, a set of known predictions (Predictive Function, PF) and that all elements of these sets are known. A question $q_i$ can then be created which covers (includes) several terms from $DF_i$ ($DF_i \subset DF$, segment of the discipline) and $q_i \in CF_i$ ($CF_i \subset CF$, questions on the segment of the discipline $DF_i$) as shown in Figure 10.

Suppose we organise a table from the elements of {DF} called Working Table, WTDF$_i$. Inside WTDF$_i$ we have overlapped areas in $DF_i$, ..., $DF_n$ ($DF_i \cap DF_j \neq \emptyset$, $i \neq j$) and where some definitions might belong to the various questions, and be “more important” than others:

$$\exists d_k \in DF_j : d_k \in DF_i, i \neq j \quad (12)$$

By organising sub-tables WTDF$_i$ for each $DF_i$ as an elementary segment of knowledge delivery – could be a lecture, seminar, etc – we can create a question $q_i$ with explanations for each term $d_k$, ..., $d_m$ from $DF_i$ as shown in Figure 11. All elements from $DF_i$ should be covered by at least one question from $CF_i$. We can then form subsets of questions {$q_i$, ..., $q_j$} to construct the full set $Q_i = \{q_i, ..., q_j\}$ where $Q_i$ covers the whole area of definitions from $DF_i$. Very importantly, the same terms from {$d_k$, ..., $d_m$} can be involved in the answers of different questions.
Once table WTDF_i is developed, the number of definitions required for the correct answer of question q_i becomes available. Furthermore, the table will also allow us to create more than one question with multiple right definitions from the same domain (DF_i), if necessary. All of them (say m) could be used together with all other possible definitions’ domains relating to DF_i. The correct answer will then mean 100% hit of right m from n. In fact, the full set of questions for one fragment of the course will form CF_i. Using other definitions from the same DF_i for the construction of different questions will actually enforce the student learning during assessment or the trial assessment exercise.

Formally, the procedure to create a question set is completed when the resultant questions cover all elements from DF_i. But this condition is not unique. We will, later on, show that there are other conditions with their own merits in forming a question set around DF_i.

The level of overlapping between working tables WTDF_i developed for a module assessment will indicate how directly connected with the structure of the curriculum they actually are. The formation of a question set, however, has both objective and subjective aspects: coverage of the module and the students’ abilities and psychology respectively. Thus overlapping of WTDF itself might be an interesting subject for further research.

7.3 Formation of the Questionnaire

Let’s assume that several questions have been built around the same subset of terms: q_i ∈ Q_i and {d_1, ..., d_m} ⊆ DF_i, and that questions from Q_i cover DF_i as shown in Figure 12.

The procedure for the generation of set Q_i is completed when all terms of the subject (or its part) are involved in the questions and, therefore, are covered by answers of some or, at least, one question. Or formally:

\[(\forall d_k : d_k \in DF_i)(\exists q_i \in Q_i : d_k \in q_i)\]  \hspace{1cm} (13)

Alternatively:

\[\bigcup_{q_i \in Q_i} q_i = DF_i\]  \hspace{1cm} (14)

If the set of questions uses several terms more than once, we can say that the subset of mostly used terms forms a core of a discipline (or its part). The rule of belonging to the core of definitions in the discipline segment DF_i is fairly simple – if there exist questions created on this segment of definitions (DF_i) and the intersection of these questions in the number of terms used from DF_i are the biggest, then these questions and elements from the segment of definitions form a core of the segment, or formally:

\[DF_i-\text{core} = \bigcap_{q_i \in CF_i-\text{core}} q_i\]  \hspace{1cm} (16)

The formation of a set of questions is the formation of CF, since these questions show how terms are connected and characterised. The selection of the core terms from DF can be combined with the selection of the core questions from CF. This double core, if determined, can be really useful for the formation of the course assessment, formally:

\[DF_{\text{core}} = \bigcup_{[i]} DF_{i-\text{core}}\]  \hspace{1cm} (17)

And:

\[CF_{\text{core}} = \bigcup_{[i]} CF_{i-\text{core}}\]  \hspace{1cm} (18)

The hierarchy of the importance of the questions can also be interesting as it helps to form an essential set of questions towards the required level of knowledge (grade) and adjust this level to different student abilities. Assume that for grade A we have one set of questions Q_A, for B we have Q_B, for C we have Q_C and Q_A ⊃ Q_B ⊃ Q_C, where A, B, C are grades of student marks. But the most important aspect of the MCAA approach is that it avoids the well known problem of MCQ, that of knowing by name. In fact, MCAA enforces students and learners to dig deeper for the meaning of definitions along with their interconnections and known predictive functions, thus pushing researchers and students to discover new knowledge – increasing the power of \{PF\}. R. Feynman [15], in his interview for the BBC emphasises the importance of such an approach, in our case for curriculum and assessment design and development: “When you finished all that naming, you know absolutely nothing what that bird is; now let us look at the bird and discover what it is doing ...”. 

![Fig. 11. Connecting questions and definitions](image)

![Fig. 12. Formation of DF_a](image)
8 MCAA efficiency

Suppose we have DF\(_i\) which has cardinality n – i.e. n terms in it and n = |(d\(_1\), ..., d\(_n\))| as shown in Figure 13. The level of their connections does not matter here.

Assume that a set of questions Q\(_i\) \(\subseteq\) DF\(_i\) which covers and includes all elements of DF\(_i\) has been built. As an example, for memory based questions in a computer architecture course, this could be: Memory: DRAM, SRAM, VRAM, Address, Data, etc. Full understanding of the meaning of the question q\(_j\) assumes a selection of right m related to the question terms, i.e. 100% correct answer for q\(_j\) is m from n possible choices. To avoid guessing by the students, the best ratio of m and n can be found, using features of binomial coefficients, to be:

\[
2 \times m = n \tag{19}
\]

This relationship between the numbers of right and wrong answers in the proposed Multiple Choice Answers Approach can be proven by considering the number of different m options from n possible answers which is determined by the classic formula of binomial coefficients:

\[
\binom{n}{m} = \frac{n!}{m!(n-m)!} \tag{20}
\]

Our task here is to find a maximum of (4) making the chances of guessing in MCAA negligible. Let’s consider two extreme examples: when m = n-1 and when m = 1. For these two variants equal results can be found:

\[
\binom{n}{1} = \binom{n}{n-1} = \frac{n!}{(n-1)!1!} \tag{21}
\]

Also, because we have n! in the upper part of the ratio in both variants it is easy to see that the minimum of \(\frac{m!(n-m)!}{n!}\) is achieved when m! = (n-m)! and, thus, n = 2m.

Suppose that n = 2a, and m = n/2, then:

\[
\binom{n}{m} = \binom{2a}{a} = \frac{(2a)!}{a!a!} = \sum_{i=0}^{a} \binom{a}{i}^2 \tag{22}
\]

For n=8 and m=4 the probability of the occasional hit on a right answer, even if the student knows that m=4, is 1/70 (1.43%) confirming the validity of this approach as a means towards excluding guessing and cheating during exams. Even semi-automatic procedures of formation of questions for assessment can decrease the workload of teachers. There is no doubt that there is much more to be done in the approbation of this approach in various universities and different disciplines.

There will, of course, be cases where the terms in the answer would be selected incorrectly and we would thus have: Selection (q\(_j\) (DF\(_i\)) \(\neq\) m. Furthermore, we may also face the case when all buttons might be pressed, i.e n from n.

For these cases a penalty function should be established, which will decrease the mark for each wrong answer and only count the right ones.

Various forms of penalty function do exist and selection of some best-fit functions can be developed even though the approach towards the selection of best-fit functions requires further research. The distribution of answers, of course, varies. The best answers on questions q\(_i\) have hit ratio 1 (m right answers from m (the same) options).

The form and severity of the penalty function is the subject of special interest and may be different whether on assessment or training purposes. A penalty function can be constructed where its effect could be “severe” in examinations, whereas one might choose another form of “kind” testing scheme for training purposes. During tutorials, for example, student should be able to experiment with MCAA. The penalty function in this case should be somehow forgiving or even soft to enable more wrong answers without visible penalisation. On the contrary, during assessment exercises the penalty function should become severe to exclude guessing. Changing the questions for each tutorial by noting which questions a student has previously answered, provides us with the facility to automatically force student to answer all questions from the discipline, and, in fact, receive essential knowledge and be assessed concurrently with learning! We feel that this feature of MCAA makes this approach promising.

Precision of the answers depends on the n and m selections and their sizes. When a candidate does not choose all right boxes for questions correctly and does not hit any wrong answers his / her total mark can be calculated by deduction of the ratio \(|m_\text{r}|/|m|\) from the total mark (equal 1):

\[
Mark \text{ in absence of wrong answers} = (1 - \frac{|m_\text{r}|}{|m|}) \tag{23}
\]

Where: m\(_r\) is the set of not mentioned correct answers.

Thus, a candidate who misses 2 from 4 right answers gets a total mark of: 1 - (2/4) = 0.5, or 50%

If the question is required to be weighted, say, cost 30 marks then the student with the sample answer above, will be awarded 15 marks.

A student, in turn, can make mistakes and may respond with incorrect answers. Again, the number of wrong answers should be weighted with the total number of wrong answers (equal to the number of right answers) and deducted from the total mark:

\[
Mark \text{ when wrong answers present} = (1 - \frac{|m_\text{w}|}{|m|}) \tag{24}
\]

Where: m\(_w\) is the set of wrong answers identified by student.
A generalized relative mark is calculated using the following set theory equation:

\[ T_m = 1 - \frac{1}{|m|} \left( |m - m_a| + |m_o \cap (n - m)| \right) \]  

(25)

Where: \( m_a \) is the answer provided by the candidate; \( m \) is the right answer, \( n - m \) is the relative complement. Note that \( m_o \cup m_a = m_o \).

Further research regarding the transition from set theory equations into arithmetic equations to calculate the grade in the assessment is needed. Formalisation of assessment procedures by use of set theory Euler, Venn or Peirce diagrams might not be so easy. Forms of penalty function must consider two variables – the number of right answers and the number of all possible answers – as these are dependent on each other. Preliminary work on these diagrams failed to convert a decision rule from set theory into an arithmetic rule in order to get a grading rule. The first impression is that the resolution of uncertainties mentioned here might be found if models for DF, CF and PF and the scheme of assessment using MCAA could be described with more than two dimensions.

9 Trial Application

A trial application of the MCAA approach to assessment in real everyday practice (module: Network Technologies) in the Faculty of Computing (FoC) at the London Metropolitan University, showed very promising results. With MCAA, students achieved much better marks than those achieved by students on the previous year when MCAA was not used either for practice or for assessment. An example of a typical MCAA question used is shown in Figure 14.

![Image](https://example.com/image.png)

Fig. 14. An example of a typical MCAA question

Statistics accumulated, an essential part of module development procedures, for the specific module by the Assessments Unit of the university supports this claim. Comparative statistics of final results from this year, using MCAA, and last year are presented in Figure 15 and show a real trend of improvement in the module results – less failures and more students achieving A and B grades – confirming that the discussed changes and the way of implementing these changes seem to be workable. The result, as a whole, looks very promising even though it is too early to discuss efficiency of separate elements in any changes planned and realised.

![Fig. 15. Statistics of MCAA approach: 2 sequential years](https://example.com/image.png)

10 Conclusions and Future Work

This work attempted to form a logical core of curriculum design for computer science and similar disciplines. Initially, the main terms used (science, knowledge, skills, curriculum and computer science) were defined and the role of education as a very important driving force in the improvement of life for the society at large was identified.

Existing models of curriculum development were briefly discussed and were found to be of a very generic nature. Their drawbacks were also identified as being applicability, usefulness and efficiency, and their measurement in the course design process. Curriculum design as information processing was then analysed and further developed by the introduction of three main functions in discipline construction: definitive, characteristic and predictive.

The core of the discipline, the way of its selection and its main features: moving and merging, were also analysed and discussed. The automatic formation of Glossary, Indexes, tracing their modifications, formation of assessment procedure and its realisation was shown to be one of the roles of IT. The two processes (moving and merging) were shown to be able to help detect changes in scientific areas related to the discipline automatically and assist the teacher towards almost automatic adjustment of the curriculum and assessment.

A new assessment methodology, Multiple Choice Answers Approach (MCAA), which apart from assessing the students also enables us to estimate the quality of teaching delivery, has been introduced and analysed. This was addressed as part of curriculum design and development, where the cycle of knowledge delivery and its progress were analysed using the same three functions: definitive, characteristic and predictive.

The features of the Multiple Choice Answer Approach assessment were analysed and the principles of the formation of multiple choice answers around a discipline, using the core of that discipline, were explained. A scheme for its realisation was shown to be through the creation of an automatic (or semi-automatic) assessment procedure and through the development of an algorithm for the selection of the most important material for assessment.

A process for the formation of an MCAA questionnaire and a scheme to maximize the efficiency of assessment, thus increasing the objectivity of the curriculum were also
introduced. Special consideration was paid on a penalty function which even though it will consider the correct answers given it will also consider the incorrect ones which, in turn, will determine the level of penalty imposed.

A trial application of the MCAA approach to assessment on a real module at the Faculty of Computing at London Metropolitan University confirmed the applicability of this methodology with the analysis of statistics showing the growth in efficiency of teaching in this specific field of computer science.

Further development of this work will incorporate the formation of a framework and design of a tool for automatic curriculum design and update using the web and any available electronic resources.

Additional investigation on some open questions regarding the structure of a discipline in terms of models of curriculum design proposed. More specifically, what does it mean … :

- … if the Index in the module is much bigger than the Glossary: \( |\{DF_u\} > |\{CF_u\}| \)  
- … if the Index in the book is much smaller than the Glossary: \( |\{DF_u\} < |\{CF_u\}| \)  
- … when the index and the Glossary are equal: \( |\{DF_u\} = |\{CF_u\}| \)  
- … when the index and the glossary are mixed, i.e. not separated: \( |\{DF_u\} \cup |\{CF_u\}| \)  

Development of the algorithm of accumulation and selection of elements for questions and glossaries as well as further extension of this algorithm towards the formation of essential questions to achieve iterative and growing understanding of the course or discipline (predictive function). Further use of set theory should formalise development of essential sub-dictionaries and questionnaires for any selected discipline.

Development of various assessment modes, including training and testing, using various supportive / advising schemes and various penalty functions – rewarding students for choosing the correct answers, even though they may not choose all the correct ones, might make assessment part of learning. Introduction and applicability analysis of special forms of penalty function for various modes of learning and testing aiming to create an applicable library, suitable for practitioners. Furthermore, the possibility of a grading function will be investigated in terms of overall level of knowledge achieved and “fine tuning” of required efforts in improving student knowledge.

Finally, the creation of an automatic framework for almost real time analysis of distributions of results obtained during the approbation period of the use of MCAA in assessment practice.

11 References