

The Design of Teaching Structure based on Competency Training of Computational Thinking

Xihua Long^{1, a}, Jiehui Zhang^{1, b}, Zhanli Li^{1, c}

¹Department School of Computer Science, Xi'an University of Science and Technology, Xi'an, 710054, China

^aemail: longxh@xust.edu.cn, ^bemail: jiehui_zhang@163.com, ^cemail: lizl@xust.edu.cn

Keywords: Computational Thinking, Teaching Structure, Teaching Element, Teaching Module, Curriculum System

Abstract. Computational thinking education focuses on the cultivation and training of a kind of thinking mode. It poses a new challenge to computing science and college computer education, including teaching contents, curriculum system, and teaching methods. For the purpose of promoting college students' computational thinking, a teaching structure is designed by restructuring the teaching contents, optimizing the employment of teaching methods, establishing compound teaching modules and a categorized and multi-layer curriculum system. Teaching practice has shown that it is helpful to the promotion of college students' computational thinking.

Introduction

Teaching structure is the framework of teaching process, which is designed for the integration of various teaching elements including contents, methods and curricular system under the guidance of certain teaching ideologies in order to achieve the established education goals. As a way of scientific thinking newly presented in the past few years, computational thinking (C.T.) has become a popular research theme both at home and abroad. Similar to mathematical thinking and engineering thinking, computational thinking should be one of the basic abilities of compound innovation talents [1]. Some globally leading colleges and universities have been trying to reform their curriculum with the aim of promoting students' computational thinking. In 2007, the US National Science Foundation launched the "Science Education in Computational Thinking" (SECANT). In 2010, the initial nine Chinese universities included in the country's "985 project" explicitly pointed out that "prominent computational thinking ability must be the fundamental skill for innovative talents in any discipline" [2].

Under such backdrop, we believe that an important and effective measure of promoting computational thinking is to design a new CT-oriented teaching structure by reexamining and restructuring teaching contents, selecting appropriate teaching methods, taking into account the college students' cognitive competence and the curriculum setting, as well as the three fundamental characteristics of computational thinking [3], namely the thinking of formative standardization, problem solving and human-machine symbiosis.

Principles of Designing Teaching Structure

A. Computational thinking

Computational thinking is a basic concept in spotlight, which concerns the essentials and the future direction of computer and computer science education. This phrase was brought to the forefront by an article on this subject by Jeannette Wing of Carnegie Mellon University. Ms. Wing defined computational thinking as "taking an approach to solve problems, design systems and understand human behavior that draws on concepts fundamental to computing"[4]. Thus the essences are abstraction and automation.

Computational thinking is not programming as narrowly defined. It includes the various abilities in the process of solving problems, such as symbolic notation, logical thinking and abstract thinking, model building, the implementation of class calculation and model calculation, as well as the use of

computer technology. It is the skill to solve problems by thinking like computers, proposing a series of viewpoints and methods, and constructing proper algorithms and programs [5].

B. Design principle

The promotion of computational thinking could help students to have a deeper understanding of the essence of calculation and the core principles of computing. It would also help to improve students' cognition and thinking in their disciplines in order to complete more creative works. Taking the promotion of computational thinking as the teaching objective of computer science education is the principle of designing new teaching structure. Its core task is to establish teaching strategies corresponding to the characteristics of computational thinking, and to establish corresponding curriculum systems according to the requirements of cultivating the ability of computational thinking.

The construction of compound teaching modules

A. The connotation of compound teaching modules

In the courses related to computing and computer science education, each knowledge point contains a number of computational thinking methods and characteristics. A compound teaching module with knowledge units and knowledge points of computational thinking is formed by the following steps: (1) to analyze and organize the teaching contents of relevant courses, to decompose them into knowledge units and knowledge points; (2) to restructure them with the orientation of computational thinking; (3) to improve the teaching contents, teaching methods and verification modes according to the requirements and standards of computational thinking.

The teaching modules are organic combinations which in the core reflect the inner connectivity of computational thinking features and knowledge points, and on the external layer guaranteed by optimized and suited teaching methods. They are a new teaching strategy for the purpose of efficient and standardized teaching. The structure of their connotation is shown in Figure 1.

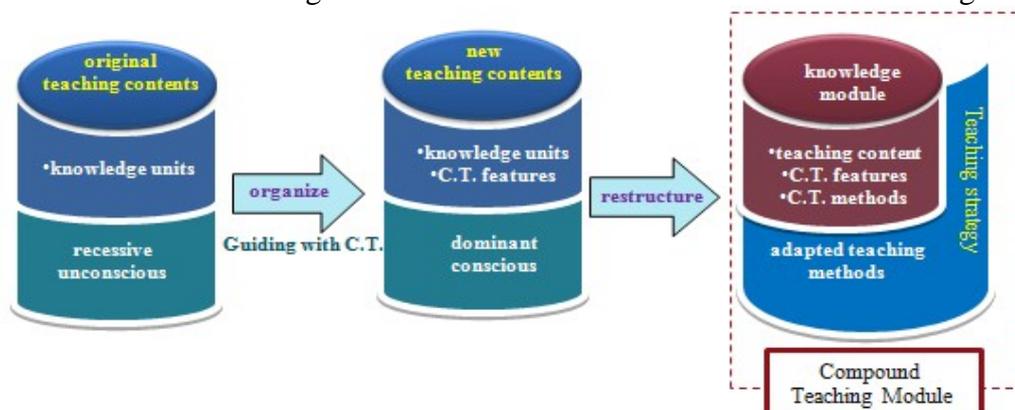


Fig.1. Diagram of compound teaching modules

B. The construction of compound teaching modules

Computational thinking unconsciously and recessively exists in current teaching contents. The “unconscious and recessive” computational thinking needs to be organized and processed into “conscious and dominant” one. For instance, in order to solve the extreme value problem in computer, the computer programme has to include simplified and transformed characteristics of computational thinking consciously and dominantly.

The basic link of thinking in the construction of compound teaching modules is to decompose methods of computational thinking into operational characteristics [6], to summarize computational thinking embodied in each knowledge point of teaching contents, to apprehend their respective roles and effects on the promotion of computational thinking, and to construct knowledge modules. In the selection of suitable teaching methods, due account should be paid to the nature and characteristics of teaching contents embodied in knowledge modules. The following specific steps are to be followed:

- 1) The “unconscious and recessive” procedures of computational thinking existed in teaching

contents are processed into “conscious and dominant” ones.

2) The teaching contents are organized according to characteristics of computational thinking embodied in them, so as to form knowledge modules which consist of knowledge points and knowledge units.

3) Suitable teaching methods are selected, with due account of the characteristics of knowledge modules and those of computational thinking embodied in them, so as to form the corresponding compound teaching module.

Take the compound teaching module corresponding to knowledge points in discrete mathematics as an example, its teaching contents, features of computational thinking, and the selection of teaching methods are shown in Table 1.

Table.1. Knowledge points in discrete mathematics and features of computational thinking and teaching methods

Knowledge Points	Concept and Method of C.T.	Teaching Methods to be Employed
symbolization of propositions	abstract , modeling	inquiring and discovering, question-driven
equivalent calculation	reduction	example analyzation, automatic learning
propositional formulas	formalization, recursion	concept attainment, inquiring and discovering
natural deduction system	reduction	inquiring and discovering, example analyzation
symbolization of compound propositions	abstract, decomposition	anchored instruction, analogy analysis

Design of Stepped Course Systems

A. The Line of Thinking in the Design of Course Systems

College students’ computational thinking is usually cultivated via a series of courses. Among the various courses, some are directed to a certain discipline, some are disciplinary, and others are general to all disciplines. In the meantime, the methods, features and level of difficulty of computational thinking demonstrated and embodied in the teaching contents vary in different courses.

The line of thinking in the design of categorized and multi-layer course systems is to formulate the cultivation requirements of computational thinking, in accordance with (1) the teaching contents, the extensiveness of utilization and the level of difficulty of methods of computational thinking, (2) the frequency of its features and the difficulty of apprehension, as well as (3) students’ ability of apprehension. Based on the above factors, courses with elements of computational thinking are then divided into different categories according to the inner connectivity of computational thinking among them. Courses in the same category are further divided into levels in accordance with the inner connectivity of course contents and the role and significance in the achievement of cultivation requirements.

For instance, Computer System course is mainly based on computational thinking, which is used to abstract and decompose complicated tasks. Numerical Analysis course on the other hand is based on both mathematical thinking and computational thinking, because both definite integral calculation (in numerical analysis) and parsing method (in mathematical analysis) evaluate integrals via the transformation of functions, but numerical solution in computer is obtained through dividing integral interval and transforming the problem into weighted sum calculation of the integrand in subinterval.

B. Modes in the Design of Course System

The essential feature of computational thinking is to solve problems in different computing environments, which reflects the different levels of computational thinking. The various courses with elements of computational thinking is divided according to the frequency of features of

computational thinking, the extensiveness of utilization and level of difficulty of its methods, the difficulty of apprehension, as well as students' ability of apprehension.

Guided by the principle of “the levels of computational thinking should accommodate the cultivation of computational thinking”, the basic methods of designing multi-layer course are as follows:

- 1) To analyze the features of computational thinking embodied in the courses, and the breadth and depth of them;
- 2) To formulate the requirements and standards of computational thinking cultivation in the courses.
- 3) To divide the courses into different categories and levels, and design a categorized and multi-layer course system.

The categorized and multi-layer course system is composed of N+X courses. The N courses of computer and computing science could be divided into “operational fundamental courses”, “operational application courses”, “algorithmic fundamental courses” and “algorithmic application courses”, which are aimed to cultivating students' operational capability and application capability. The X interdisciplinary courses are “engineering application courses” which are aimed to cultivate students' capability of research and innovation. Multi-layer course system is shown in Figure 2.



Fig.2.The categorized and multi-layer course system

For instance, in categorized course system, College Computer Fundamentals and courses of computer language (such as C Language, FORTRAN Language) are categorized as “operational fundamental courses”; the likes of VC++ Programming, Web Programming and Java Programming are categorized as “operational application courses”; Numerical Analysis, Optimization Method, Data Organization and Database Application are categorized as “algorithmic fundamental courses”; while Mathematical Modeling, Artificial Intelligence and Pattern Recognition are categorized as “algorithmic application courses”; and lastly Application of Numerical Calculation in Mineral Engineering, Numerical Simulation in Underground Projects and Computer Geology are categorized as “engineering application courses”.

C. An Example of Course System Design

In the cultivation of computational thinking, the computing task needs to be described in a stable, formalized and unitary language. Programming language, as a determinate data system, could be used to train formalized thinking. Therefore, the purpose of “operational fundamental courses” is to cultivate students' ability of transforming natural thinking into computational thinking. This category of courses is featured by the standard form of computational thinking.

Take Mathematical Modeling, a course which combines mathematical thinking and computing thinking, as another example, the most important step is to build mathematical models and solve them. To solve mathematical models via computer is to describe the process of problem solution through programmed and mechanized methods. Therefore, the purpose of “algorithmic enhancement courses” is to cultivate students' implementation capability of transforming

mathematical thinking into computing thinking, enable them to raise and solve a problem by computational thinking. Computational thinking is featured by the thinking of problem solving.

A categorized and multi-level course system based on the setting of disciplines and majors at Xi'an University of Science and Technology is demonstrated in Table 2.

Table.2. The categorization of courses in the course system

Course	Categorization	Level	Requirement of Computational Thinking
College Computer Fundamentals	Operational Application Course	1	abstraction and formalization, simple problem solution
Java Programming	Operational Application Course	1	abstraction and formalization, problem solution
Numerical Analysis	Algorithmic Fundamental Course	1	transformation of thinking, process design
Design and Analysis of Algorithms	Algorithmic Fundamental Course	2	transformation of thinking, process design
Principles and Application of Data Digging	Algorithmic Application Course	2	establishment of objects, alteration of process, verification of establishment
Artificial Intelligence	Algorithmic Application Course	3	establishment of objects, strategy planning, alteration of process, verification of establishment
Computer Geology	Engineering Application Course	2	transformation of thinking, establishment of objects, strategy planning, problem solution, verification of establishment

Conclusion

A compound teaching model organically integrates the essential features of computational thinking, teaching contents and teaching methods. The categorized and multi-layer course system reflects the various levels of computational thinking and the different requirements for the cultivation of computational thinking ability. This new education structure that consists the compound teaching models and categorized and multi-layer course system, which is oriented by computational thinking. It studies and targets those courses of computing science and computer education; also it is aimed to transform knowledge education to the cultivation of thinking. Most importantly, it lays a good foundation for the achievement of promoting college students' ability of computational thinking.

References

- [1] GL.Chen, SG.Dong. Computational Thinking and Basic College Computer Education [J]. China University Teaching, 2011, 1(7) 7-11.[in Chinese]
- [2] C9: Joint Statement on of Computer Basic Teaching Development Strategic [J]. China University Teaching, 2010, 241(9) 4-9. [in Chinese]
- [3] ZW.Xu, DD.Tu. Three New Concepts of Future Computer Science [J]. Journal of Computer Science and Technology, 2011, 26(4) 616-624.
- [4] J. M. Wing. Computational Thinking [J], Communication of the ACM, 2006, 49(3) 33-35.
- [5] Q.Mou, L.Tan. Research and Progress in Computational Thinking [J], Computer Science, 2011, 38(3) 10-15.[in Chinese]
- [6] J.M.Wing. Computational Thinking and Thinking about Computing [J]. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 2008, 366(1881) 3717-3725.