Research of Computational Thinking-driven Teaching and Innovative Practice Pattern

Zhanquan Wang, Jiang Liu, Chunhua Gu, Qingchun Hu, Xinxiu Wen
Computer Teaching & Experiment Center, East China University of Science and Technology, Shanghai, China

Abstract - With the improvement of computer technology, the previous methods of teaching and innovation practice aren’t conformable in current information age. It is difficult for students to understand computational thinking (CT) and improve their innovation ability. The study propose a new pattern of computer course system based on computational thinking, which consists of modules, including computational thinking module, foundation teaching module, innovation practice module, ACM train module and information literature module. The results of the study indicate that the new teaching system is very effective and validate. Student engaging in this study achieve good performance in the application ability contest of the computer in China and Shanghai city.

Keywords: Computational Thinking; Innovative Module; Information Platform;

1 Introduction

Computer technology and computer knowledge is becoming increasingly popular. For the instructor, how to make the students use computer to create the professional product, how to popularize computational thinking and how to make students truly use computer to accelerate the cultivation of innovation ability, is a long-term problem for computer education. At the second national university undergraduate teaching working conference, Ji Zhou, the previous minister of education, pointed out that "We must strengthen teaching work and improve teaching quality." [1] One of the major work is to vigorously strengthen the cultivation of college students' learning ability, practical ability and innovative ability. According to the characteristics of the current basic course system, Guoliang Chen [2] [3] who is academician of the Academy of Science of China, introduced the concept of computational thinking to apply the basic concepts of computer science to problem solving and system design, and a series of thinking activities about computer science [11]. Academician Chen penetrates computational thinking and creative ability penetrate into the architecture of computer basic education and makes the ordinary students really grasp how to do work and create more schemes with computer through basic computer teaching mode change, which aims to cultivate students with computer teaching and innovation practice.

2 Current computer education teaching

Currently, the university computer basic courses prefer more basis and focus on credits, less application and ignore capacity. After the completion of the university computer basic course [4] [5] [6], such as fundamentals of computers, programming, application courses, most of non-computer professionals courses for students tend to the end, without the support of follow-up professional practical and innovational course system. The students learn professional courses directly, the computer knowledge learned is rarely practical to solve practical problems, especially for the professional areas. The computer is not combined well with the no-computer profession due to the lack of thinking about how to think for the computer. Although the non-computer professional undergraduate will use some computer knowledge when doing graduation design, however, the students' computer application ability need be further improved because of a period of no practical time. In addition, students also have some other problems in the process of learning computer knowledge, such as the teaching just for teaching without using innovation and practice exercises, and don't grasp of computational thinking deeply. For students, they don't know how to use what they learn, which does not reflect the ability of the computer to solve problems. Due to the existence of these phenomena, teachers feel that so much basic computer knowledge can’t be well pass on to students, and students can’t get a good absorption, the key spirit of innovation and innovation practice projects is neglectful. This situation restricts the development of the students' computer knowledge, and affects the application and expansion of basic computer knowledge in the field of non-computer professional, which is not conducive to cultivating innovative talents for the whole country. With the development of information technology, the computer is an essential tool, like the invention of the printing in our country, as we can hardly move an inch without computer. So it is extremely important to foster the thinking and innovative ability of students. Response to these phenomena, firstly we start from the teaching mode set of basic computer course. The purpose of building basic tutorial system about developing computational thinking innovative capacity is to promote and foster the students’ ability of using computer to improve the level of expertise and innovative ability.
After discussion and literature review, our computer basic teaching system mode has great changes. Figure 1. shows our earlier basic computer teaching system model. Initially, this mode greatly promotes the development of basic computer teaching. However, it has encountered some problems under the rapid development of computer technology. The following starts from architecture for next analysis.

<table>
<thead>
<tr>
<th>Computer application curriculum</th>
<th>electronics commerce</th>
<th>practical netwro technology</th>
<th>3D animation technology</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>computer technology curriculum</td>
<td>database technology and application</td>
<td>multimedia technology and application</td>
<td>software development and application</td>
<td>system and network technology</td>
</tr>
<tr>
<td>program design group</td>
<td>C language</td>
<td>JAVA language</td>
<td>VB</td>
<td>Web</td>
</tr>
<tr>
<td>basic curriculum group</td>
<td>college computer basis course</td>
<td>college computer elementary course</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Figure 1. Teaching curriculum group architecture](image)

The architecture is divided into four levels: basic curriculum group, program design group, technology curriculum group and the application curriculum group, and does not reflect the concept of computational thinking. In the lessons, students prefer to learn and apply, rather than analyze and solve problems from a higher level. The information interaction and knowledge sharing of the course is not perfect. In this case, the pattern of course architecture based on computational thinking is proposed, which takes computational thinking as the main line. In addition to the co-ordinated development of various modules, the construction of information-sharing platform will help a variety of knowledge sharing.

3 Teaching Innovative Practice Platform of Computational Thinking driven

According to our situation and the development of computer technology, in conjunction with other universities [2] [6] [7] [8], the original architecture has been improved. As shown in Figure 2: 1) Huge homework in the course, where it is a combination of the feature of no-computer professionals and practical projects related to the curriculum, such as the students development the system related chemical engineering simulation. At the end of terms, students should complete the homework; 2) Take the conceit of computational thinking as the driver, and make the coordinated development of various modules, while focusing on training students’ innovation ability; 3) Optimize original architecture of the basic course group t; 4) Add or replace some courses to make the computer basic course group more reasonable, such as C#, VB.NET, Python, and network attack and defense etc..

![Figure 2. Computational Thinking-driven Teaching and Innovative Practice architecture](image)

The improved architecture is mainly divided into teaching system module, innovation practice base module, computational thinking driving component, ACM training, and information teaching platform module. Furthermore, the computer basic course group architecture should be taught for every course managed by Computer Teaching & Experiment Center, which mainly aims at talking about preliminary knowledge about computational thinking, and also making students know the place where the basic computer course they are learning in the course architecture. So that they should know what they have learned, what they will learn and which computer practical innovation and intramural competitions they can participate in. Each part will be described in the following.

3.1 The opening chapter of the computer architecture

The opening chapter: non-computer science students often feel quite confused about what they can learn on basic computer courses. They also have no idea about what teachers should talk about, either how to solve the problem from a computer perspective. Furthermore, they are confused about the level where they are and the use of it? Therefore, the teachers need to tell students that the position of the course they are learning, what they can do when they eventually finish them, the importance of computational thinking and the things we can do in conjunction with professional knowledge and how to conduct innovative practice activities with the use of computational thinking. After this guide, the students will put some loss and know the level of computer knowledge they have learned and what they need to do, how to use computational thinking to solve the problems in the applications. All in all, the opening chapter is very important, because we have to make the non-computer science students know that what we are training is, the future computer talents of non-computer science, the creative talents using computational thinking to solve problems in this field.
3.2 Strengthen Computational Thinking Efforts and Improve Computer Connotation Construction

This module emphasizes the idea of computational thinking can make students better grasp it, for the drive platform of primary computational thinking (showed in Figure 2). We should strengthen the understanding of the basic computer concepts through experiment and practice to improve the computational thinking ability of students.

Firstly, we must understand the definition of computational thinking, be conscious of what computational thinking and computer knowledge connotation are. Secondly, we try to understand the connotation of computational thinking, because of computational thinking far more than meaning for computer programming, but also requires the ability to think in abstract multilevel. Thus, to develop computational thinking is to broaden everyone's way of thinking, open the ideas of everyone. Especially, to the classroom or innovative practice, we request: first of all, teachers should often lead students to think, conduct more and teach more how to analyze and solve practical problems with the computer perspective. For example, how to solve (using mathematical equations) it from the perspective of mathematical thinking and with computational thinking (using iterative approximation), how to use the computer to solve it is an application of computational thinking. Secondly, teachers should also put forward questions appropriately and talk more about original computer knowledge to make students know how the computer is to solve the problem use computer, then it can strengthen students' abilities and interests of computational thinking. In specific practices, we will evaluate the ability of computational thinking through construing and lectures. However, also less the research of the quantitative assessment for computational thinking. So we will study and design system to evaluate computational thinking and connotation construction in this project, further to specific the evaluation of computational thinking to the key points of teaching, to make quantitative assessment and to improve connotation construction, accordingly to make it more scientific through researching and improving the project, and form an iterative rising spiral of continuous improvement process.

3.3 The Optimization of Innovative Practice Base Modules

In Figure 3, we have done some appropriate changes for innovative practice base modules. Due to the strong practicality of computer knowledge, if we only learn computer theoretical knowledge, what students master is not well. Therefore, according to the university’s requirements, we make some further reform for innovative practice module, such as increasing the cultivation of computational thinking, letting students know how computer works and why it set such a variable rather than as set. Thus, it allows students to improve computational thinking skills, and facilitates them better conduct computer innovative practices; Another example is that increasing excitation measures and leading into improvements based on computer-interest, including the primary intermediate and senior innovative practice activities, ACM and the knowledge of a variety of computer knowledge contests etc.. At the same time, we evaluate innovative practice base construction quality through innovative practice performance (such as student evaluation, competition, etc.).

3.4 Teaching Curriculum Group architecture

In teaching architecture module in Figure 2, the professional and enhancing classified guidance teaching could be talked. For example, students are encouraged to participate in the innovative practice activities in the form of huge homework, the specific method is to mix lesson comprehensive training (practice activities) into course groups, allow students to freely compose a group of 2-3 to finish a characteristic professional engineering work, and then guide them into the intermediate practice and innovation of practical activities according to the completion situation, so as to lay a foundation for the next senior innovation activities and cultivate good seedlings. In addition, because of the majority of high school students have access to computers, so we detect whether students need to learn the course of college computer elementary course through the examination. Taking the learning interest and the combination with professional practice into account, we will introduce C#, VB.NET, python and other courses. In the practice, we assess the teaching quality by lectures, student interview, test scores and the projects. For both the humanities and science, we take classified teaching to better enhance students' learning interest and enthusiasm. Specific description is in Figure 3. Simultaneously, in the classroom, we also explain the application method of computational thinking, particularly:1) in program design group, process-oriented variables, find and sort(type, recursive, optimization and compromise);data structure(abstract, recursive, compromise decomposition);expressions, control statements, function (abstraction, recursion, decomposition, reduction); System building, code reuse, and module building (system recovery, planning, parallel, simulation, protection). 2) in network course, due to the variety of the different nature of network courses, we will explain different content depending on different levels. In general, we will talk about: finding transmission problems, such as transmission error (retransmission), CRC(protection, redundancy, error correction and system recovery); the OSI hierarchical structure and the TCP/IP structure (reduction, decomposition, scheduling, SoC, etc.) network protocol / function (abstraction, decomposition, compromise etc.) etc..
3.5 Computational Thinking Driver

This part is very important under the teaching of primary computational thinking, the students have had some ideas about the computational thinking. On this basis, this part will focus on how to use the knowledge of computer commonality. For example, for the 20 key concepts of computational thinking (including calculation, communication, coordination, automation, design reduction, embedding, transformation, simulation, recursive, parallel, abstract, decomposition, protection, redundancy, fault tolerance, error correction, system-recovery, heuristics, scheduling). For different students, we use different emphases and strategies when explaining specifically, such as ACM programming contest, we will explain all of it; for junior students, we will explain communication, coordination, automation, design, reduction, embedding, transformation, simulation, recursive, parallel, abstract, decomposition, protection and redundancy; for other innovation practices and teaching, we will explain some concepts selectively according to the characteristics of the course. For example, in C programming course, firstly, we will explain the concept of computational thinking, and require students to review literature about computational thinking. Then, we will ask students to show their understanding and reflection of computational thinking with regard to the knowledge points of each chapter, and request students to give examples of their use in real life. Finally, they should complete a project which is relative to real life. The activities of students can be learned in the teaching platform through background management functions. In the aspects of practical application, interested students are encouraged to submit their own design jobs, then further to participate in innovative practices in computer contests at all levels. The practice shows that the method works well, it is worth further improved.

Additionally, in the ACM construction driven by computational thinking, in connection with the characteristics of ACM students (we have counted for each recruiting, more than 90 percent of students are proficient in C language and data structure), in this case, we will let them master more computational thinking concepts, while obtaining preliminary computational thinking by themselves. We emphasize on instilling the 20 technical points of computational thinking at thinking training platform and then apply it into various practical modeling. Such as focusing on coordination, design, reduction, embedding, transformation, simulation, parallel, abstract, decomposition, protect, redundancy, fault tolerance, error correction, heuristics and scheduling, some other basic concepts can be acquired through self-study. By way of emphasis training, we obtained very good results in the ACM contest, which is shown in the table 1. At the same time, ACM team members act as teaching assistant role, they can help us organize various computational thinking lectures and train student computational thinking ability. For example, for the minimum cost maximum flow analysis, the minimum spanning tree problem, etc., which involves reduction transformation, simulation, recursion, type, SoC, planning, scheduling, compromise, redundant, fault tolerant, heuristics, etc., they will explain well to innovation team members and the other groups.

3.6 The Information Teaching Platform

An information technology teaching platform is developed due to the diversification of communications between teachers and students in our university. The platform includes homework upload system and forum (including innovative practice sub-forums) etc.

4 Practice and conclusions

By the improvement of computer teaching architecture based on computational thinking, the peer teachers of many universities such as Tong Ji university, ShangHai university, East China Normal University etc approve our work. We made great progress in the aspects of teaching group architecture, and strengthen students' computational thinking ability and practical innovation. After the implementation of the new teaching innovation platform, the level of theory and practice of our students greatly increase, especially computational thinking ability. Students can learn to analyze and solve problems from the perspective of computer, and these improvements directly reflect into the practice innovative module. After the teaching of innovation practice, our students' ability has been greatly improved, the mutual development of theory and practice levels directly promote the improvement of the level of innovative practice, and achieved good results. For example, in the past year, more than 300 people have participated in computer knowledge contest. In classroom and innovation practice, our teachers use computational thinking ideas to impel students to analyze and solve problems from computer perspective with computational thinking issues, and achieved very good results. For instance, in Shanghai computer application ability contest in 2010, 2011 and 2012, we have achieved very good results (table 1); in Chinese college students computer design contest, we have won many prizes, and we have obtained the outstanding organization award in Chinese college students computer design contest in 2012; Since our students possess a high level of theory, computational thinking and practical level, the East China University of Science and Technology also achieved very good results in ACM finals is also obtained.


<table>
<thead>
<tr>
<th>years</th>
<th>quantity of students enrolled in practise</th>
<th>quantity of student passed(the credit is 0.5)</th>
<th>Shanghai’s prize (the first prize)</th>
<th>Nation prize(th e first prize)</th>
<th>ACM’s prize(gold medal)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009-2010</td>
<td>800</td>
<td>305</td>
<td>3 (1)</td>
<td>1 (0)</td>
<td>There are three Gold medals, seven silver medals, sixteen bronze medals.</td>
</tr>
<tr>
<td>2010-2011</td>
<td>1000</td>
<td>360</td>
<td>12 (2)</td>
<td>3 (1)</td>
<td></td>
</tr>
<tr>
<td>2011-2012</td>
<td>1200</td>
<td>532</td>
<td>13 (2)</td>
<td>8 (1)</td>
<td></td>
</tr>
</tbody>
</table>

All in all, the new computer teaching platform has achieved good results. At the same time, computational thinking drives to increase innovation practice, which greatly improved the students' enthusiasm, initiative, practical ability, not only make students master computer knowledge well, but also deepen the mastery of basic computer knowledge through theory teaching, practice, competition and innovation practice and other methods, so that students can better use computational thinking ideas to think about how to use computer tools to solve problems in the real world. Thus it can better trains students’ abilities to identify problems, solve problems. But each system mode has two sides, that whether it will bring good results in the implementation process needs to be adjusted according to the actual situation of our university and local conditions. Only in this way can it achieve the best effect.

5 References


