The Research on Curriculum Group of Computer Programming Based on CDIO

Xiaodong Huang, Qin Wang, Lishang Zhu  
Capital Normal University, Beijing 100048, China

ABSTRACT: Computer Programming and related courses account for a large proportion in current information courses. Such courses typically include C/C++ language programming, VB language programming, Java language programming, C# language programming, data structures and algorithms. There are many common aspects in terms of knowledge points among programming courses. However, currently, the courses are taught separately. Furthermore, under the trend of shrinking time allocation and declining number of courses in undergraduate curriculum, it is critical to maximize the use of limited hours of programming courses to enhance teaching effectiveness. Based on inquiry-based teaching model in accordance with CDIO teaching philosophy, it is proposed in this paper to conduct programming curriculum group construction. Therefore, knowledge of program design courses within the group will be shared and courses instruction within the group will be coordinated. Knowledge sharing facilitates its application to experiments and projects, which can improve students’ comprehensive capabilities, especially their ability to solve practical problems.

KEYWORDS: CDIO; Computer Programming; Curriculum Group; Inquiry-based

1 INTRODUCTION

CDIO engineering education philosophy was put forward by the research team from the Massachusetts Institute of Technology, Chalmers University of Technology, Linköping University and the Royal Institute of Technology, after four years of exploration and research. Accordingly, the international organization named after CDIO was also founded [1-4]. CDIO stands for Conceive, Design, Implement and Operate. Traditional classrooms are teacher-centered. Teachers impart knowledge orally to students with the help of lesson plans. Students just passively listen to lectures. By no means did the main role of students reveal. Differing from the traditional discipline-based teaching methods, CDIO teaching philosophy emphasizes active learning. Putting learning in the complex, meaningful problem situations, learners take initiative in solving problems via independent research and corporation. Thus, students can learn hidden science behind the issues and form problem-solving skills and the self-learning ability.

Programming requires a wide range of knowledge of students, especially their logical thinking ability. Students may encounter many problems. At present, in programming courses, emphasis has been laid on elaborating programming syntactical rules, which is not conducive to the cultivation of students’ programming abilities. Therefore, in programming courses, importance should be attached to active exploration of students. They are supposed to utilize programming languages to develop software and solve practical problems, which, in turn, can further promote students’ mastery of the programming concepts.

Currently, four years’ undergraduate education requires a lot of class time. Various colleges and universities are striving to relieve the burden of students, so that students can have more self-study time. Therefore, it becomes increasingly important to make full use of the limited teaching hours. Students majored in computer sciences are required to attend 4-5 programming courses, so we think there is urgency for programming curriculum group construction. Programming course barriers should be overcome to the greatest degree. In the limited class time, the common problems of programming courses should be addressed. CDIO teaching philosophy should be adhered to while lecturing programming courses, which will strengthen students’ inquiry learning. Hence, students’ programming literacy and logical thinking ability will be enhanced, which will improve the overall teaching effectiveness of programming courses.
2 PROGRAMMING CURRICULUM GROUP CONSTRUCTION

Computer programming curriculum group construction is to integrate program-related courses in accordance with CDIO philosophy. Reformed will be carried out in teaching organization, content, structure in light of professional goals. As a result, there will be a clearer teaching purpose, more prominent training orientation, and more continuity in skill development process. At the same time, the extension courses of programming curriculum construction will also be continuously expanded.

Computer programming lectures concentrate on cultivating students’ programming ability. Students can accumulate programming experience and master programming languages. By means of programming practice, students can obtain solid basic skills and will take initiative in exploring unknown problems using acquired knowledge.

In our teaching practice, we listed all knowledge points covered in programming courses and fused them to form a new programming class, and named it programming cluster courses. The cluster courses fall into three teaching categories: programming clustering curriculum basics, programming clustering curriculum advanced programming skills and algorithm, programming clustering curriculum practice.

2.1 Programming clustering curriculum basics

Sequence structure, selection structure and cycle structure are elaborated in C language, VB language and the Java language. So we integrated these elements and lectured in one course while constructing curriculum group. Moreover, we compared and contrasted the similarities and differences of the three structures in the 3 languages. The limited class time was fully utilized, and simultaneously, it is convenient for students to master the basic structure of the three languages. Object-oriented programming is elaborated in C ++ language, Java language and C # language, which can also be placed in one lecture and taught simultaneously. Meanwhile, the development environments of VB language and Delphi language are both visual integrated development environment, which can also be grouped and taught together.

2.2 Programming clustering curriculum advanced programming skills and algorithms

Sequence table and chain table are crucial concepts in data structure. While learning C ++ template, its implementation technology will be described, which is to gain new knowledge by reviewing the old. Another example is the string concept in the C language. String class appears in C ++, which can realize more complete functions. Moreover, it is easier for students to accept these functions, which will contribute to deepen their understanding. Students can achieve a string class by themselves so in their future programming it can be called directly. In addition, repetitive knowledge points will be saved to languages with more comprehensive functions. For instance, file operations will not be lectured in C language and students can self-study. While teaching C ++, file operations will be explained. Thus redundancy will be avoided.

Table 1. Course hour allocation of programming curriculum group basics, programming techniques and algorithms, and practice.

<table>
<thead>
<tr>
<th>Course</th>
<th>Basics</th>
<th>Programming techniques and algorithms</th>
<th>Course practice</th>
<th>Percentage of practice in total hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>C language programming</td>
<td>10</td>
<td>22</td>
<td>32</td>
<td>50%</td>
</tr>
<tr>
<td>Java language programming</td>
<td>10</td>
<td>22</td>
<td>32</td>
<td>50%</td>
</tr>
<tr>
<td>C++ object-oriented programming</td>
<td>10</td>
<td>22</td>
<td>32</td>
<td>50%</td>
</tr>
<tr>
<td>Data structure</td>
<td>10</td>
<td>32</td>
<td>22</td>
<td>34%</td>
</tr>
<tr>
<td>C# object-oriented programming</td>
<td>10</td>
<td>22</td>
<td>32</td>
<td>50%</td>
</tr>
<tr>
<td>VB language programming</td>
<td>10</td>
<td>22</td>
<td>32</td>
<td>50%</td>
</tr>
<tr>
<td>Delphi language programming</td>
<td>10</td>
<td>22</td>
<td>32</td>
<td>50%</td>
</tr>
</tbody>
</table>

2.3 Programming clustering curriculum practice

Computer programming is highly practical. Students can absorb theoretical knowledge through practice experience and strengthen their programming abilities. Theoretical lectures and practical classes should be allocated rationally. Course practice is generally divided into curricular and extracurricular programming practice; specific hour arrangement is shown in Table 1.

Curriculum group can reduce redundant experiments. As to the content, “VB Programming” focuses on the interface effect; “C Language Programming” focuses on the basic knowledge of grammar and common algorithms; “Data Structure” focuses on structures and algorithms, as well as the characteristics of different structures; “C ++ , Java and C # Object-oriented Programming” focuses on design and implementation.

In terms of contents arrangement, as basic knowledge shared by these languages are taught in programming clustering courses basics, we can reduce the proportion of basic knowledge in
3.1 Programming ideas to pass through this course to students. Programming. Engineering education can be a good CDIO can be regarded as various stages of Conceive, design, implement and operate entailed by programming knowledge from Internet or books in order to finish these practical projects, which will further cultivate the students’ self-learning ability and creative ability. In order to achieve seamless link between courses, extracurricular programming originates from the content of the course, but it should be more difficult than the curriculum.

3 PROGRAMMING TEACHING MODEL BASED OF CDIO CONCEPT

Conceive, design, implement and operate entailed by CDIO can be regarded as various stages of programming. Engineering education can be a good idea to pass through this course to students.

3.1 Conceive and design

3.1.1 Classroom organization
The classroom teacher is no longer merely in charge of explanation. Instead, about half of the time will be used for checking whether students have grasped programming knowledge by means of inviting students to offer comments or completing practical work. The questions put forward and discussions stimulated allow students to explore and develop ideas, find errors, and sum up the rules. Teachers will spend less time on explaining the new knowledge point and they will just guide students to understand the new knowledge, which can lay a solid foundation for the following practice content. There is no need to cover every aspect of programming language and grammar. Teachers will just concentrate on new ideas and corresponding programming methods. Students will further their understanding of new knowledge and fully grasp the details of the grammatical rules in course practice.

3.1.2 Teaching evaluation
Currently, there are mainly three types of teaching evaluation: student self-assessment, in-group assessment, assessment by members of different groups. Taking into consideration the implementation plan of programming curriculum group, three aspects of evaluation can be identified: the first is the assessment of students’ basic knowledge, which can be evaluated according to the testing results and process; the second is the assessment of students’ advanced programming skills and algorithm, testing students’ proficiency level of programming skills; the third is the assessment of students’ course practice. Students will be randomly selected and attend the defense. Students will be ranked in accordance with their defense performance and project progress report.

3.1.3 Engineering ability
The nature of engineering is creation and production. CDIO is the concentrated generalization and abstract expression of “learn by doing” and “project-based education and learning”. Students familiarize themselves with the life cycle of projects engineering (including products, production processes and systems), ranging from research and development to operation. Thus, they acquire and master engineering skills in an active, practical way, with the emphasis on the organic link among the curriculum.

3.1.4 Creative ability
The teaching model represented by inquiry-based instruction can promote active learning, which is conducive to cultivating innovative talents.

3.2 Implement
As For the implementation of the curriculum, the author of this paper takes “C Language Programming” as an example and elaborates how to implement the curriculum based on conceiving and designing courses.

3.2.1 Classroom instruction
Increase the proportion of programming practice in classroom instruction. Invite experienced developers from corporations home and abroad to collaborate with teachers to guide students in their project planning and practice. In specific curriculum practice, one student will be responsible for summarizing the main content and difficulties of the project before practice; simultaneously, another student will be in charge of preparing a programming example. The other students will discuss and analyze the example in class; ultimately, the instructors will give their comment. This teaching approach will train students in terms of summarization, analysis and articulation, which will further enhance the development of their innovative ability.

3.2.2 Individual programming
Each student needs to be able to learn and master the corresponding programming methods and techniques. This requires students to deepen their study on actual algorithms and programs on the basis of classroom instruction. First, students must read a large number of exemplar programs; secondly, students modify and improve these exemplar programs by means of imitation; finally,
students will be competent in programming and practice.

3.2.3 Teamwork programming
Programming teams are formed when each student has mastered some basic programming knowledge. Team capacity training will focus on communication, coordination and collaboration between the team members. In order to develop these capabilities, students will be divided into several study groups. Group members can discuss and communicate with each other and accomplish their shared goals. Teachers will offer assessment of and feedback to each group’s cooperative learning.

When grouping students, various factors should be taken into consideration. A reasonable method is to assign students with different programming capabilities into one team. In the specific programming assignments, each group can be formed roughly by 4-6 students, a group leader will be elected, who should have strong programming skills and communication abilities. Team members also need to play an active role in exploring specific projects practice. They should be able to plan and offer suggestions on project progress for discussion.

3.2.4 Team progress summary
On the basis of teamwork, each team must regularly submit summary reports. The group leader will convene a meeting once a week, in which everyone needs to report on the progress of their own module and raise issues if they face any. Group members will discuss and offer reasonable suggestions. Then, on this basis of the discussion, each person will write the summary report on their own module. Finally, the leader is responsible for compiling the team progress summary report on the basis of the reports by group members.

3.2.5 Course assessment
As for programming curriculum group assessment, emphasis should be laid on practical programming capabilities assessment especially on the assessment of process. As to programming basic knowledge and programming skills and algorithm, students will be examined via the programming flow chart, programming algorithms and algorithm realization. As to programming practice, students will be assessed via the completion of each module by team members, project schedule planning and execution capabilities, data retrieval capability, project progress reports and the ability to compile the final report.

3.3 Operate

3.3.1 Students are encouraged to take the initiative in identifying problems and forming groups, developing and solving problems.
CDIO concentrates on enhancing students’ engineering skills, a very important part which is to identify and solve problems in the process of project implementation.

For instance, after studying “Java Language Programming”, students found that Java language can be applied to the development of Android cell phone. Students tend to download some applications to the smart phone APP. Then some students found that certain APP software cannot meet their special needs. It occurs to them that they can capitalize on their acquired knowledge to develop new applications. Teachers encourage students to form cooperation teams and jointly develop APP based on their respective interests and strengths.

3.3.2 Students are encouraged to make full use of the network and learn new programming techniques.
CDIO encourages horizontal comparison of and link formation of different courses. Software programming requires students to integrate knowledge of many courses and apply it to practice. Thus, students should not only master a variety of programming languages, but also understand and grasp the latest application and programming related thoughts such as data structure, algorithms and design patterns. Students should consciously form the schema of a variety of programming knowledge during programming projects practice. They can learn together the useful knowledge points which are of great use in their programming but not covered by the teacher in class by means of group discussion.

For instance, while learning C++ language, some students are interested in how to define the dynamic array. Teachers in turn encourage students to search related programming methods and techniques on the Internet. This kind of teaching methodology facilitates students’ self-learning ability and autonomous learning ability in the actual practice of programming.

4 CONCLUSIONS

Programming curriculum group construction based on CDIO teaching philosophy is proposed in this paper. This mode attaches importance to programming knowledge sharing within the curriculum group. Emphasis will be laid on collaborative instruction while teaching courses within each curriculum group. Students’ capabilities will be enhanced comprehensively, namely,
consolidating their basic knowledge concerning programming, improving their utilization of advanced programming techniques and algorithms and ultimately successfully completing practical projects. The teaching mode in accordance with CDIO teaching philosophy strives to achieve the integration of teacher instruction and students practice and the integration of individual learning and team cooperative learning. Programming practice can effectively strengthen communication and cooperation between students and teachers. Meanwhile, it also contributes to develop team spirit and enhance students’ ability to cooperate.

5 ACKNOWLEDGEMENTS

This work reported in this paper is supported by Capital Normal University teaching reform project “CDIO-based program design teaching model reform and practice”.

6 CORRESPONDING AUTHOR

Huang Xiaodong, dawn_hxd@yeah.net

REFERENCES