Teaching Reform and Practice Based on OBE Concept

Taking "Computer Integrated Course Design" as an Example

Hongmei Fan(✉), Yumin Wang, and Jingwen Chen

School of Energy and Environment, Southeast University, Nanjing, China
101006992@seu.edu.cn

Abstract. Based on the outcomes-based oriented teaching education (OBE) concept, a task-driven teaching implementation plan was designed for the environmental engineering with combination with the teaching characteristics of "Computer Integrated Course Design" course. The student-oriented teaching mode was carried out, the teaching effect was evaluated, and the improvement measures and suggestions were put forward to realize the transformation from quality monitoring to continuous improvement.

Keywords: Outcomes-based education (OBE) · Environmental Engineering · Computer integrated program design · practice

1 Introduction

Under the guidance of outcomes-based education (OBE) concept, college teachers should actively promote the teaching reform, design talent training programs scientifically and reasonably, and strive to achieve the transformation from discipline orientation to goal orientation, from teacher-centred to student-centred, and from quality monitoring to continuous improvement in the process of talent training [1]. To this end, the teaching group of "C++ course"in environmental engineering of Southeast University launches the teaching reform practice of "Computer Integrated Course Design" under the background of new engineering.

2 Curriculum Training Objectives

The "Programming and Algorithmic Language" course, which belongs to the advanced computer language programming course, is one of the basic computer application courses for the first-year students in the university of science and engineering. At present, the teaching of C++ language is still based on Dos operating system, while the application software was developed based on Windows operating system. As such, it is particularly necessary to introduce "Visual C++ programming" as the teaching content of "Computer Integrated Course Design" on the basis of "C++ Programming and Algorithmic Language".

Y. Yau and F. Hage Chehade (Eds.): DESD 2022, ASSEHR 691, pp. 94–98, 2023.
https://doi.org/10.2991/978-2-494069-37-4_14

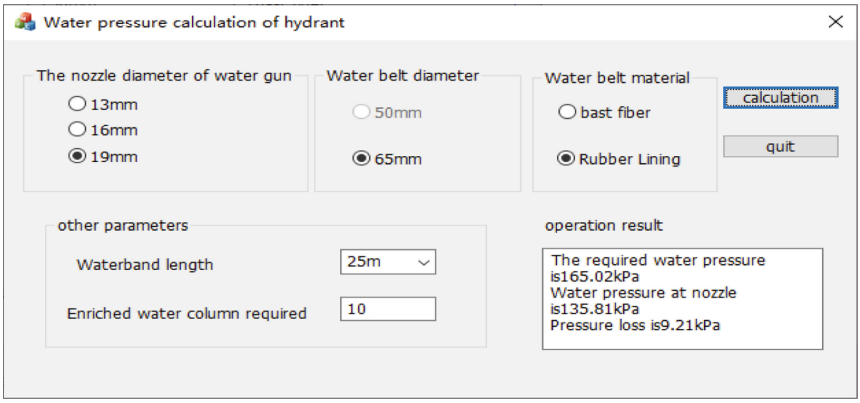

Fig. 1. Example of Original Application Program Interface

The overall goal of Computer Integrated Course Design is to consolidate the theoretical knowledge and operation skills of C++ language and improve the comprehensive application ability of C++ programming, master the basic method of developing application program, and cultivate students' ability of using C++ language and programming knowledge to solve engineering application software design.

3 Design of Task-Driven Teaching Implementation Plan Based on OBE Concept

3.1 Teaching Model and Methodology Design Based on OBE

3.1.1 Subjects no Longer Depend on Textbooks

The teaching time of "Computer Integrated Course Design" is relatively short (including theoretical 8 h and experimental 32 h), which requires students to master Windows programming. However, due to the large scale of C++ programming complex structure, and Microsoft Foundation Class (MFC) provides a large number of application framework code, and strong professionalism, many beginners feel very difficult to study. In order to enable students to grasp the principle of Windows program operation and program writing process in a short time, the task-driven teaching method was proposed.

After investigation, it was found that many colleges and universities are highly dependent on teaching materials in the teaching of "Computer Integrated Course Design". The topics in the teaching materials are mainly information management, Such as library management system, train booking system, student information management system, etc.[2] These design topics have poor professional relevance and lack of innovation.

3.1.2 Task-Driven Teaching Mode

Reasonable introduction of tasks to ensure clarity, feasibility and operability of task objectives. Freshmen in grade 1 have not yet entered professional courses, and the easy problem is that task setting is too difficult. The combination with professional knowledge

is too difficult, which has a negative impact on students' learning enthusiasm. Therefore, the accumulated original application programs related to professional knowledge were used as the driving target to combine task-driven with programming courses. "Building Water Supply and Drainage Engineering" is an upcoming course for sophomore majoring in environmental engineering. Students are relatively familiar with the layout of indoor sanitary equipment and water supply and drainage pipelines. In addition, the relevant design departments of building water supply and drainage engineering can provide many ready-made original applications (in Fig. 1). Therefore, we choose to combine the specialized course with "Computer Integrated Course Design".

Define the task and decompose the task concretely. According to the contents of "building water supply and drainage engineering" textbook chapters (domestic water supply system, drainage system, fire water supply system, rainwater drainage system, and hot water supply system, etc.), the teaching task was decomposed, and each chapter can be used as an independent task point. Students select an interesting chapter to learn knowledge points, extract useful information, and clarify the task of curriculum design.

3.2 Student-Led Teaching

3.2.1 Student-Centered Teaching

At present, the C++ language taught by the university is based on the Dos operating system. It is a sequential process-driven programming method. Each program has obvious start, execution process and obvious results. Visual C++ is a Windows-based programming method, which is an event-driven (non-sequential) programming method. When students learn Visual C++ programming, there is a big step to jump, which is the difficulty of learning Visual C++ programming [3]. In order to make students smoothly realize the transition of the two programming methods, the following three steps from C++ language to Visual C++ was designed, which is expressed as follows.

1. Solving C++ window-oriented programming.

Windows program running principle and program writing process, written by C++ language WinMain function, window initialization function, message processing function and window process function, expounds the Windows program running mechanism.

2. Solving C++ window-oriented simple application box. The operation mechanism of WinMain function in MFC program; the role of global variable the App; common API function call methods.
3. Solving MFC AppWizard automatic generation framework. Examples are given to illustrate the differences between MFC and C++ design habits; the definition of variables and the difference between data transmission and traditional C++ programming are introduced. Method of adding entity classes in MFC; MFC class library functions corresponding to each control in the dialog box are used.

3.2.2 Arrange Progressive Tasks According to Individual Differences of Students

Due to the differences among students, there are also significant differences in learning conditions. In order to meet the learning needs of different students, so as to achieve the purpose of personalized customization learning, course design tasks can be divided into general requirements, improved requirements and higher requirements of self-selection. All students should be able to meet the general requirements, most students should be able to improve the requirements, some students meet the higher requirements of self-selection, in order to achieve stratified diversion teaching [4].

4 Evaluation of Teaching Effectiveness

According to the overall ideas and measures of the above education reform, the teaching reform practice was carried out. The practical effect is as follows: Firstly, the difficulty of the topic is moderate. Secondly, 90% of the students can complete the task well, but there are also 10% of the students only complete the basic function of the application. More than 50% of the students can do the same type of problems one by one, and independently complete similar engineering applications. About 10% -20% of students solve problems from a more innovative perspective and can challenge highly difficult applications. Finally, with a clear curriculum goal of solving professional problems, students' learning enthusiasm is greatly improved.

5 Conclusions

Based on the OBE concept, we should actively promote the education and teaching reform to achieve the transformation from quality control to continuous improvement. The following improvement measures were put forward for the teaching reform practice of the course "Computer Integrated Curriculum Design". Firstly, the programming practice should not only emphasize completion, but also emphasize quality, so as to cultivate students' ability to write high-quality code. Secondly, record a video on the implementation process of the "Computer Integrated Course Design" to showcase the intermediate links such as group discussions, code debugging, teacher presentation. Finally, engineering ability and innovation ability need to be obtained through a lot of practice, the cultivation of "computer integrated curriculum design" ability should be realized in the subsequent professional courses.

Acknowledgments. The paper is funded by "Research and Practice Project of Labour Education Teaching Reform in Southeast University in 2021".

References

1. Zhiyi Li, Hong Zhu, Zhijun Liu, etc. Guiding the teaching reform of higher engineering education with the concept of results-oriented education[J]. Higher engineering education research, 2014, 2: 29–34 (in Chinese).
2. Li Qian. C Programming Practice Teaching [J]. Modern Information Technology, 2018, 2 (10):145–148 (in Chinese).
3. Rong Li, Xiangdong Qin. Teaching reform of task-driven programming course [J]. Technological wind,2021, 9:42-44 (in Chinese).
4. Haojie Chen. Teaching practice of ladder method in Visual C++ programming course [J]. Electronic technology, 2021, 50(08): 44-45(in Chinese).