Research and Practice on the Teaching Mode of the Course Based on the OBE Concept-Taking the C Programming Course as an Example

Meijiao Zhang and Youwei Xie
Quanzhou University of Information Engineering, Quanzhou, China

Keywords: OBE Concept, C Language, Teaching Mode, Course Objectives.

Abstract: C language programming is an important professional foundation course for computer science majors, and the teaching content mainly includes the syntax of C language and programming algorithms and ideas. Based on the OBE teaching concept, the objectives of the C language programming course are formulated based on the requirements for the training of software engineering majors, and the teaching reform and practice are carried out in terms of teaching contents, teaching methods and means, and course assessment methods, which improve students' learning enthusiasm and practical ability, and also provide a reference solution for the teaching reform of other programming courses.

1 INTRODUCTION

The Internet era no longer restricts access to knowledge by time, space and geography, schools are no longer the exclusive agents of knowledge and the era of big data has brought with it vast and abundant educational resources. Yet motivation and attention to learning have become scarce resources. In order to stimulate and mobilise students' enthusiasm and motivation for independent learning, and to change passive learning into active learning, it is necessary to fully implement the student-centred concept, to reform the traditional teacher-centred teaching methods into mixed and diversified teaching methods, and to change a single paper examination into a comprehensive assessment. The OBE output-oriented, student-centred and continuous improvement education concept has therefore been increasingly recognised by educators and universities (Zhao Bingtao, 2019).

OBE (Outcome Based Education) is a model of education in which student outcomes drive the operation of the education system (Li Zhengya, 2022). OBE emphasises 'student-centredness' and focuses on what outcomes are achieved by the educated and how to effectively help students achieve them. All teaching and learning-related activities, including the curriculum, teaching and learning methods, and assessment, should be designed so that all students can successfully achieve the desired learning outcomes at the end of their learning.

Taking the study of the C programming course by students of the Software Engineering major of the School of Software of Quanzhou Information Engineering College in Class 2021 as an example, the course leader leads the course team to determine the teaching objectives of the course that meet the graduation requirements and facilitate assessment, to formulate a teaching plan and syllabus that meet the positioning of the school and can support the achievement of the graduation requirements, to review whether the teaching contents and the course assessment methods can achieve the teaching objectives and whether The course content and assessment methods are reviewed to ensure that they can achieve the teaching objectives and emphasise the development of students' abilities, and to determine the assessment cycle of the course. Teachers implement the course according to the syllabus and collect all data during the course. At the end of the course, the course evaluation team proposes improvement measures in response to the problems identified during the teaching and evaluation process, and forms a record document to adjust the teaching plan and optimise the teaching assessment and evaluation criteria (Wu L, 2022).

Zhang, M. and Xie, Y.
Research and Practice on the Teaching Mode of the Course Based on the OBE Concept-Taking the C Programming Course as an Example.
DOI: 10.5220/0012287100003807

In *Proceedings of the 2nd International Seminar on Artificial Intelligence, Networking and Information Technology (ANIT 2023)*, pages 531-537
ISBN: 978-989-758-677-4

2 DEVELOPMENT OF THE OBJECTIVES OF THE C PROGRAMMING COURSE UNDER THE OBE EDUCATION MODEL

The graduation requirements for students are set in the professional training programmes of universities, which generally involve professional and engineering skills, interpersonal skills, teamwork and communication skills, professional abilities and attitudes, and complex engineering skills. These graduation requirements are supported by the curriculum system, which forms a matrix to support the graduation requirements of the talent training programme. The course objectives are designed to reflect students' learning outcomes, to be clearly aligned with the graduation requirements, and to guide the teaching and assessment of the course (Xiaofei Sun, Xiong L). The C Programming course in our software engineering program is supported by Graduation Requirement 1 Engineering Knowledge, Graduation Requirement 3 Designing/Developing Solutions and Graduation Requirement 5 Using Modern Tools. In response to these requirements and the characteristics of C programming, the course team members, after much discussion, finally identified three course objectives for the C Programming course. Objective 1: To gain proficiency in the basic syntax knowledge of the C language. This includes basic language knowledge of programs and programming languages, algorithms and algorithm descriptions, standard data types and expressions, common library functions, statements, constructed data types (arrays, pointers, structures, commons, enumerations, etc.), etc.; Objective 2: To be able to develop good programming habits and styles, to form certain self-learning and debugging error abilities, problem analysis skills, code reading skills, and teamwork skills. Be able to write C programs and design experimental solutions based on practical problems; Objective 3: Be familiar with the running process of C programs, be able to proficiently configure the C program development environment and use programming tools, and be able to consult relevant standards and manuals.

Combined with the course objectives of C programming and the corresponding graduation requirement index points for software engineering majors, the course team has determined the correspondence between the two after several studies and discussions, as shown in Table 1.

Table 1: Table of course objectives and supporting relationships to graduation requirements.

Graduation requirements	Graduation Requirements Indicator Points	Support strength	Course Objectives
1. Engineering knowledge	Indicator Point 1-3: Be able to apply software engineering expertise and tools to reasonably simplify, reason and analyse complex software engineering problems, build mathematical models and solve them.	H	Objective 1: Basic knowledge of the syntax of C, including basic language knowledge of programs and programming languages, algorithms and algorithm descriptions, standard data types and expressions, common library functions, statements, constructed data types (arrays, pointers, structures, commons, enumerations, etc.), etc.
3. Design/develop solutions	Indicator Point 3-3: Be able to design, develop, manage and maintain complex software projects and demonstrate a sense of innovation.	H	Objective 2: Be able to develop good programming habits and style, develop some ability to learn and debug errors on their own, problem analysis, code reading, and teamwork. Be able to write C programs and design experimental solutions based on practical problems.
5 Use of modern tools	Target 5-1: Understand the principles and methods of using software development, testing and debugging support platforms and IT tools commonly used in software engineering-related fields, and be able to understand the limitations of modern tools in solving practical problems.	M	Objective 3: To master the basic ideas and methods of structured programming, to be familiar with the running process of C programs, to be able to configure the C program development environment and use programming tools proficiently, and to consult relevant standards and manuals.

3 TEACHING IMPLEMENTATION OF C PROGRAMMING COURSE

In line with the principles of student-centred education, the course should be organised and implemented in such a way that it is focused on student learning outcomes, student development and student learning (Niu H, 2022). The C Programming course team has carried out a detailed design of the teaching content, teaching methods, course assessment and evaluation methods, as follows.

3.1 Teaching Content, Allocation of Credit Hours, and Support of the Course Objectives by Teaching Content

In order to better achieve the course objectives, the course team has carefully selected knowledgeable and interesting programming cases as classroom teaching cases, and set up a set of topics from easy to difficult for each chapter; carefully designed student experiments to effectively improve students' programming skills, logical thinking ability and innovative thinking. The teaching content and allocation of teaching hours and the corresponding course objectives are designed as follows: the course is divided into nine teaching units, namely C language overview, C programming fundamentals, selection structure programming, circular structure programming, arrays, functions, pointers, structures and other data types, and files. Course Objective 1 is supported by Teaching Units 2, 3, 4 and 5; Course Objective 2 is supported by Teaching Units 3, 4, 5, 6, 7 and 8; Course Objective 3 is supported by Teaching Units 1 and 9. The allocation of credit hours to each teaching unit and the relationship to the support of the course objectives are shown in Table 2.

Table 2: Table of teaching content, allocation of hours, and the relationship of teaching content to support the course objectives.

Teaching content	Support for teaching objectives	Classroom teaching			Online self-directed learning
		Credit hours	Lectures	Experiment	
Teaching Unit 1 Overview of the C language	3	4	2	2	1
Teaching Unit 2 Fundamentals of C Programming	1	8	4	4	2
Teaching unit 3 Choice structure programming	1, 2	6	4	2	2
Teaching Unit 4 Loop Structure Programming	1, 2	10	4	4	2
Teaching Unit 5 Arrays	1, 2	10	4	4	2
Teaching Unit 6 Functions	2	8	4	4	2
Teaching Unit 7 Pointer	2	8	4	4	2
Teaching Unit 8 Structures and other data types	2	6	4	2	2
Teaching Unit 9 Documentation	3	4	2	2	1
Integrated projects	3	0	0	4	0
Total hours		64	32	32	16

3.2 Reform of Teaching Methods and Tools Based on OBE

Based on the OBE student-centred and result-oriented concept, the course team has reformed the teaching method around the course objectives, changing from the traditional teaching mode to a flipped classroom + online and offline hybrid teaching mode, in which the online teaching resources are the provincial-level high-quality online open courses built by the course team on the Super Star platform, and the experimental platform chosen is the PTA programming class experimental teaching aid platform. PTA is a series of national programming courses teaching team of Zhejiang University, in cooperation with Netease and Hangzhou Baiteng Education Technology Co., Ltd, launched in September 2015 for universities and society, automatic assessment of the program, open teaching aid platform, committed to building a quality teaching resources for schools and enterprises to build and share the platform (Lian Xinze, 2022). At present, there are 991 universities, 8319 teachers, more than 250,000 high-quality questions and more than 3.6 million students registered. Teachers can choose questions already in the question bank or create their own test questions for students to practice. The system not only allows students to evaluate the correctness of the program immediately after programming, but also allows them to check their ranking in the class, and if they are able to do so, they can also do the fixed set of questions in the system, in which they can see their ranking among the national university students. The course has been designed separately for before, during and after class, as shown in Figure 1.

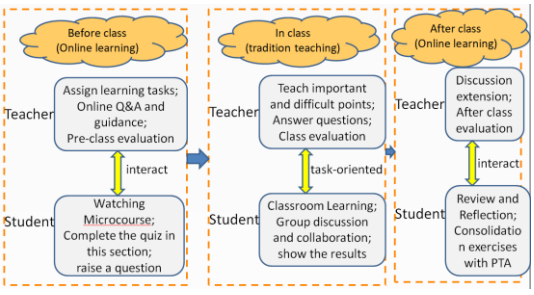

Figure 1: Flipped classroom + online/offline hybrid teaching design.

1) Before Class

The teacher uses Study Link to assign learning tasks and pre-class reflection questions so that students can clarify their learning objectives and

content for the next class. Students complete the learning tasks set by the teacher, which include learning the textbook, online courseware and videos, and completing online quizzes. Students can leave messages to the teacher via StudyPass or QQ when they encounter difficulties in the learning process. Teachers can use the test results to grasp students' pre-class learning in order to determine and adjust the content of classroom lectures. 16 lessons in total.

2) In-Class

There are two types of offline delivery: classroom and laboratory, both of which are described below.

Classroom teaching, divided into two types, 16 sessions in total, 32 hours

① Lecture + small flip

Based on students' learning and questions collected from the web platform before class, teachers deepen their understanding of students' knowledge through quick questions and answers, explain key points and difficulties with examples, and summarise (Honglei M, 2022). In class, interesting and knowledgeable programming problems are assigned for students to discuss in small groups and write algorithms. 2-3 groups of students are invited to present their programs or algorithms at a time, and the teacher comments on them appropriately or students evaluate each other.

② The Great Classroom Turnaround: My Classroom, My Talk

The instructor of this course will assign integrated design projects in the middle of the term through Study Pass, and students will work in groups to collaborate, discuss and work together to complete them outside of class time. In the last two sections of the course, a representative from each group will present their program on stage and run a demonstration, which will be critiqued by the teacher. All students other than the group representative will present on stage on a sample basis, and each group will submit a project design report at the end (Zhu Y, 2022).

Experiments - a combination of practice and lectures, online and offline, 16 sessions in total, 32 hours

Two segments are included:

① On-board practicals, students log on to the PTA platform to complete the practice problems assigned by the teacher.

② Flip the experiment and the teacher will observe students completing the exercises. 15-20 minutes are left at the end of the experiment for students to assess and comment on each other's mistakes.

3) After School

Students brush up on their practice on PTA and teachers post on QQ about students' completion of exercises and answer questions with low correctness. Teachers post some extended knowledge of the course through blogs to improve students' information literacy.

Through the reform of teaching methods, students' interest in learning and programming ability have improved significantly. Students are often seen working overtime to brush up their skills, because they can see the results of other students, they will silently compete with each other, so they show a state of catching up with each other. Most of the students are active in class, their programming skills have improved significantly, many of them ask the teacher to assign harder topics, and the failure rate at the end of the term has dropped significantly. In addition, our school is an application-oriented university, which focuses on cultivating students' practical application ability. In order to meet the training objectives of our university, we have organized teachers who teach C language programming to compile a textbook, which has been published by Tsinghua University Press in 2018 and has been used in teaching. The textbook emphasizes the combination of theory and practice. A large number of examples to solve practical problems are compiled in the textbook, and a matching exercise solution and computer experiment course is also written. Corresponding to our self-compiled teaching materials, we organized teachers to make multimedia courseware with the teaching materials. In the usual teaching activities, we organize teachers to carry out teaching research activities in view of the problems in teaching, such as: Pointers that students find difficult to learn and other chapters, and study how to break through the difficulties. Through self-compiled textbooks, multimedia courseware and teaching and research activities, the level of all teachers has been improved.

3.3 Method of Assessment and Weighting of Courses

In order to better fulfil the course objectives, this course designs the assessment content separately according to each course objective. The course adopts a combination of online and offline, phased, multiple cumulative and marathon learning assessment and evaluation mode, in which students' academic status and other assessments are conducted before or during the teaching process, and a summative assessment test is conducted centrally at

the end of the period, in order to keep abreast of students' existing knowledge level and the progress and problems they have made during the teaching process, so as to timely adjust and improve the teaching programme and promote students' development. The mid-term and final examinations of the course will be conducted using an online testing platform instead of paper-based papers to improve the fairness and objectivity of the examinations. The course objective assessment consists of online learning situation, midterm exam, final exam, experiment and comprehensive project grades, with the following weighting: 0.2 weighting for objective 1 online learning situation, 0.3 weighting for midterm exam (10 points for question 1 and 20 points for question 2), 0.5 weighting for final exam (10 points for question 1 and 20 points for question 2); 0.2 weighting for objective 2 experiment (including experiment 6-8), 0.2 weighting for midterm exam (25 points for question 25 marks for Question 3, 45 marks for Question 4) weighted 0.3, final exam (20 marks for Question 3, 20 marks for Question 4, 30 marks for Question 5) weighted 0.5; Objective 3 experiments (containing experiments 1-5) weighted 0.5, integrated project design weighted 0.5, as shown in Table 3 below.

Table 3: Table of assessment methods and weightings for each course objective in C programming.

Course Objectives	Assessment methods	Weighting	Basis of evaluation
Course Objective 1	Online Learning	0.2	Online learning
	Mid-term exams	0.3	Questions 1-2
	Final exams	0.5	Questions 1-2
Course Objective 2	Experiment	0.2	Experiments 6-8
	Mid-term exams	0.3	Questions 3 and 4
	Final exams	0.5	Questions 3-5
Course Objective 3	Experiment	0.5	Experiments 1-5
	Integrated projects	0.5	Integrated project completion

3.4 Evaluation of the Achievement of the Objectives of the C Programming Course

The course objectives are mainly evaluated quantitatively, and the data source is the assessment of each assessment link in the course syllabus that supports the course objectives. If the course objective i is supported by N assessment methods, assuming that the total score of the jth assessment method is Sj, the average grade of students' assessment is Aj, and the weight is Wj, the assessment value OAi of the achievement of course objective i can be calculated as shown in equation (1):

$$OA_i = \sum_{j=1}^{N} W_j \frac{A_j}{S_j} \quad (1)$$

The sum of the weights of all the assessment points in equation 1 that support the achievement of course objective i is 1 (Zhifeng Yang, 2022).

A total of 117 students of C language programming in the class of 2021 were assessed, and the lowest achievement rating value of the three objectives was 0.73 and the highest rating value was 0.78, so the achievement rating value of C language programming course objectives was 0.73. The assessment results of students were divided into five levels, and the distribution of students for each achievement degree of each course objective is shown in Figure 2.

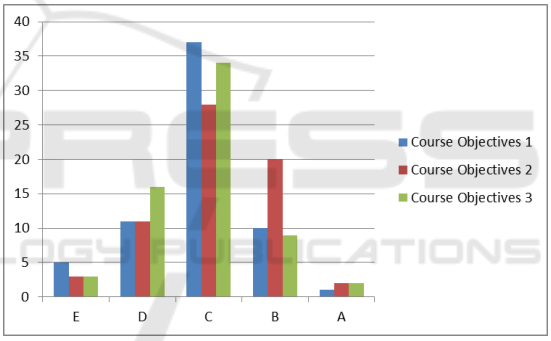

Figure 2: Distribution of students' achievement of C programming course objectives.

The graph above shows that the distribution of students' overall performance is largely normal, with all four course objectives being achieved in the class as a whole, as assessed by an expectation of 0.70, but a total of 22 individual students did not meet the standard. Of these, Objective 1 and Objective 3 were completed satisfactorily, with over 85% of students having an expectation value greater than 0.7, and Objective 2 was completed at an average level of 82% achievement. From the C programming course objectives and corresponding teaching content, students have a good grasp of the basic knowledge of this course, but the comprehensive application ability is insufficient, and most students are unable to use functions, pointers and structures flexibly for programming.

3.5 Continuous Improvement of C Programming Course Teaching

The attainment rating for Course Objective 1 was 0.76 and the distribution of students' scores was largely concentrated between 60 and 90, relatively scattered, with two students in particular scoring low and away from the concentration zone (Luo J, Wu Xiusheng). The improvement measures are that special attention needs to be paid to individual students with poorer knowledge and ability, and individual tutoring can be provided when necessary to slowly improve students' knowledge and application skills according to their needs. At the same time, the average of all students in Objective 1 is not high and there is still room for improvement. In subsequent teaching, the main focus needs to be on improving the overall students' basic knowledge level and ability. The attainment rating for course objective 2 was 0.73, and the distribution of students' scores was basically concentrated between 60 and 90, which was relatively scattered and distributed over a wide range, with four students scoring low and far from the concentration area. In the subsequent teaching, attention was paid to the learning of the lower-achieving students and teaching was tailored to their needs. Also the average score for students in Objective 2 was not high, with many students being more inflexible in algorithm design and lacking in programming knowledge and programming skill levels. The class attainment rating for Course Objective 3 was 0.78, with three students scoring relatively low and with a low overall score. Students' ability to synthesise and apply their knowledge is inadequate and their patience and motivation need further strengthening. According to the assessment, the marks of the experimental design and final assessment show that there are still some students who fail to follow the teacher's teaching progress in time and need to adjust their teaching methods and progress (Li Zonghua, Limin T). In the future, there are a number of suggestions for continuous improvement in the teaching process:

① Focus more on the development of students' comprehensive application skills, pay attention to the learning of lower-achieving students, and tailor the teaching to their needs.

② Continue to strengthen the development of students' practical skills, comprehensive design and creative abilities.

4 CONCLUSION

As China's information technology continues to deepen, the country needs more high-quality software engineering professionals. c programming language, as the first programming language in our school, is an important professional foundation course, and also a course with strong practicality and application. The reform and implementation of OBE-based C programming is not only an innovation of the traditional teaching mode, but also a significant and long-term work . The course is based on the "student-centred" principle, which emphasises both theoretical teaching and practical skills training. In the process of implementing the curriculum, it is necessary to set teaching objectives, strengthen the content of the curriculum, build teaching resources, reform teaching methods and means, and set reasonable assessment methods, etc. Although these issues increase teachers' knowledge and teaching workload, the curriculum reform allows students to effectively learn new knowledge, master new methods, develop new skills and solve new problems, and become the biggest beneficiaries of the curriculum reform. These are also the goals that we as educators seek to achieve. Teaching is an endless road, and we will continue to explore it.

ACKNOWLEDGMENTS

This work was financially supported by Fujian Province Young and Middle-aged Teachers Educational Research Project "Research on Network Intrusion Detection Technology Based on Deep Learning"N0.JAT220456.

REFERENCES

Zhao Bingtao, Cui Guomin, Wu Jun. Teaching reform of the course based on OBE concept in the context of the new engineering discipline: An example of the course "Air pollution control engineering"[J]. *Journal of East China Institute of Technology (Social Science)*, 2019, 41(04):376-379+385. https://doi.org/10.13256/j.cnki.jusst.sse.2019.04.014.

Li Zhengya, Zhang Lin. The construction and practice of university English teaching system based on the concept of "OBE": the case of Xi'an College of Arts and Sciences [J]. *Journal of Xi'an University of Arts and Science (Social Sciences Edition)*, 2022, 25(04):83-86.

Wu L. Research on the Construction of "Trinity" Teaching Mode of Eap Reading for Non-English Major

Postgraduates Based on OBE --A Case Study of Postgraduate English Course [J]. *Curriculum and Teaching Methodology*, 2022, 5(15). https://doi.org/10.23977/CURTM.2022.051515.

Xiaofei Sun, et al. "Exploration of Data Structure Teaching Mode Based on OBE Against the Background of New Engineering". Proceedings of the 8th International Conference on Economics, Management, Law and Education (EMLE 2022).Ed., 2022, 98-104. https://doi.org/10.26914/c.cnkihy.2022.073686.

Xiong L, Li D. Principles and Methods of Online and Offline Blended Course Construction Based on the OBE Concept[C].*Proceedings of the 3rd International Conference on Education Studies: Experience and Innovation (ICESEI 2022)*.ATHENA PUBLISHING,2022:42-46. https://doi.org/10.26914/c.cnkihy.2022.073155.

Niu H, Fan J, Yan H. Research on the Integration of Subject Competition and Professional Course Teaching Mode Based on OBE Concept [J]. *Frontiers in Educational Research*, 2022, 5(16). https://doi.org/10.25236/FER.2022.051604.

Lian Xinze, Zou Dong Dong, Li Jing, et al. Teaching reform of programming course based on spelling A [J]. *Educational Teaching Forum*, 2022 (33):57-60.

Honglei M, Yinfei W. Exploration of Online and Offline Blended Teaching Reform Based on SPOC+Mu Classroom--Taking Product Innovation Design and Development Course as an Example [J]. *Frontiers in Educational Research*, 2022, 5(21). https://doi.org/10.25236/FER.2022.052104.

Zhu Y, Zheng S. Research and Practice of Online and Offline Blending Teaching Mode of “Foundation of Program Design”[J]. *International Journal of Learning and Teaching*, 2022, 8(4). https://doi.org/10.18178/IJLT.8.4.213-218.

Zhifeng Yang, Dongming Nie, Long Li, et al. Weighted evaluation algorithm of curriculum goal attainment under the concept of Professional certification [J]. Journal Of Hengyang Normal University, 2022, 43(03):38-42. https://doi.org/10.13914/j.cnki.cn43-1453/z.2022.03.015.

Luo J, Xiao S, Chen L. Research on the evaluation system of practical Teaching Quality based on OBE concept [J]. *Computer Informatization and Mechanical System*, 2022, 5(1).https://doi.org/10.12250/JPCIAMS2021090403.

Wu Xiusheng, Cao Jufang, Wang Aiguo, et al. Project-driven reverse design of professional practice course system [J]. *Journal of Hubei University of Education*, 2020, 37(08):72-76.

Li Zonghua. Analysis of OBE-based evaluation of software engineering course goal attainment [J]. *Computer Knowledge and Technology*, 2022, 18(25):151-154. https://doi.org/10.14004/j.cnki.ckt.2022.1678.

Limin T, Xikun L, Jiangping Z. Java Programming Course Reform Based on OBE Concept under the Background of New Engineering Education [P]. *Distance Education and Learning*, 2020.