

Research on Experimental Teaching Strategy of C Language Public Course under the Background of New Engineering

¹Liu Dongxia and ²Zhang Yongbo,

¹School of Computer Science and Technology, Shandong University of Technology, Zibo, China

²Foundation Department, Zibo IT & Engineering School, Zibo, China

Abstract—This paper analyzes the current situation and shortcomings of experimental teaching in the public course of C language programming for non computer majors under the background of new engineering. By analyzing the relationship between confirmatory experiment, comprehensive experiment and design experiment, this paper makes a preliminary discussion on how to improve the methods of experimental teaching and how to improve the effect of experimental teaching.

Keywords: *New Engineering, Ability, Training, Stepped, Design, Code Specification*

I. INTRODUCTION

In order to promote the development of China's hard science and technology, enhance international competitiveness and national hard strength, and cultivate innovative engineering and scientific and technological talents, the Ministry of education actively promotes the construction of new engineering subjects, and issued «Notice on carrying out research and practice of new engineering», «Notice on promoting new engineering research and practice projects», «Opinions on accelerating the construction and development of new engineering and implementing excellent engineer education and training plan 2.0», etc, to explore the new engineering education system.

Non computer science and engineering students in our university offer a public courses in C language programming in the second semester of freshman year. This course mainly cultivates students' computational thinking ability, data analysis ability, problem solving ability and basic programming ability. As the program design course is a highly theoretical and practical course, if students want to learn this kind of course well, on the one hand, they should complete the learning of classroom knowledge points with high quality, on the other hand, they should also carry out more computer practice. Therefore, strengthening the management and guidance of curriculum experiment can effectively improve students' practical ability.

There are some problems in the traditional experimental teaching mode. First, many students are not fully prepared before practicing on the computer, do not know how to solve the program errors, and rely too much on teachers to solve the problems. Second, the content of the experimental course is not professional, and there is no systematic explanation on the characteristics of programming. Students can only "draw a ladle according to a gourd" during operation, and will not draw inferences by analogy. Third, students pay more attention to function realization and ignore code programming norms in the process of experiment. The specific manifestations include

random naming of variable names, unclear code layout, poor program readability and so on.

In view of the above problems, I adopted the progressive training mode of paying attention to the cultivation of thinking mode, clarifying the code writing standard and the cooperation of various experimental forms, and achieved good teaching results.

II. CULTIVATION OF THINKING MODE

C language experiment teaching needs to cultivate students' Computational Thinking, algorithmic thinking and systematic thinking. Computational thinking refers to using the basic concepts of computer science to solve problems, design systems and understand human behavior. Algorithmic thinking is not the repeated recitation of some excellent codes that have been designed, but the practice of their own abstract ability to solve problems, that is, the ability to code or design programs to solve problems from abstract problems. Systematic thinking is a kind of thinking method that people use systematic view to systematically understand all aspects of the object and its structure and function.

To cultivate students' way of thinking, it needs to be carried out in several steps from easy to difficult. Firstly, in the teaching and experimental practice of sequential structure programming, cultivate students' understanding of basic C language grammar rules and sequential structure program execution process, be able to use various formulas to solve simple calculation problems, and preliminarily establish the concept and thinking method of programming, so as to strengthen calculation thinking. Then, through the experiment of selecting structure, loop structure, array and pointer, guide students to transform the real problem into the representation in the computer, reasonably select and organize data according to the programming requirements, and complete the algorithm design. So as to deepen the thinking of algorithm. Finally, through the experiments of function and file, guide students to establish modular design ideas and solve complex problems from complexity to simplicity. Finally, I can complete a small project to cultivate students' systematic thinking.

III. CODE SPECIFICATION

This course has formulated a series of requirements for program code writing. From the first class, students can feel the beauty of program writing and that writing standard source code can greatly improve the readability and maintainability of the program. These writing requirements form a special C language code writing specification document, which is placed on the school's network teaching platform for students to read, check and compare at any time.

Taking DEV C ++ editor as an example, the main contents of the writing specification include:

(1) The overall style should be neat. Specifically include: braces separate branches; Various clauses must be indented; For multi operator mixed operation, appropriate parentheses shall be added, and a statement shall be written on one line as much as possible.

(2) Naming of identifiers. The name of any variable, constant, function and structure shall clearly reflect its purpose and meaning. It is recommended to use short English words or Pinyin.

(3) Add comments to the function block. Specifically: use double slash // to add comments to the function header to briefly describe the functions of the function. Use the double slash // to add comments to the structure to describe the purpose of the structure and the meaning of its members. Use the double slash // to add comments on a module that implements some functions to describe the purpose of this part of the code.

(4) Common errors. Specifically, it is prohibited to use variable values without initial values; Prohibit the use of uninitialized pointers; Floating point numbers of different types cannot be compared equally; Goto statements are not recommended; Try to avoid mixed operations between negative numbers of signed integers and unsigned integers.

IV. LADDER TRAINING OF PROGRAMMING ABILITY

Learning is like climbing steps, which is a process of gradual improvement, and the design of experimental tasks should also follow this law. The whole course is designed with 7 experimental tasks, with increasing difficulty, taking into account basic training and ability training. See Table 1 for details.

Table 1: Ladder training schedule

Task	catalogue	Specific contents:
Task 1	Sequential structure programming	Case conversion; Fahrenheit temperature and Celsius temperature conversion; Inverse positive integer;
Task 2	Branch structure programming	Output the maximum of three integers; Sort three integers in ascending order; Calculate the value of piecewise function;
Task 3	Loop structure programming	Output all prime numbers in the interval; Count the number of letters and numbers; Output isosceles triangle composed of asterisk;
Task 4	Application of array	Output the first N items of Fibonacci sequence; Output the maximum and minimum values of N integers; Calculate the sum of two m-row and n-column matrices;

Task 5	Use of pointers	Use the pointer to output the array elements in reverse order; Use pointer inversion to output array elements; Use the pointer to find the specified data in the array;
Task 6	Custom function	Write a function to determine the prime number and call it in the main function; Calculate the maximum value of N numbers with function;
Task 7	File operation	Input and output of data in the file;

CONCLUSION

The goal of the construction of "new engineering" is to cultivate a large number of high-quality compound talents with strong engineering practice ability, strong innovation ability and international competitiveness. After several semesters of practice, it has been proved that the teaching mode of combining the cultivation of thinking ability with the ladder cultivation of programming ability has achieved good results. Students have less fear of difficulties and more spirit of active exploration in programming courses. The educational environment is changing, the level of students is changing, and the teaching methods have been improving.

References

- [1] The Ninth National Congress of China Association for science and technology was held in Beijing [EB/OL]. (2016- 05-30)[2020-11-02]. www.gov.cn/xinwen/2016-05/30/content_5078085.htm
- [2] Li Wei, Shen Xiaoling, Wang Lei, etc Exploration and practice of agile Teaching[J] computer education. 2019(9): 129-133.
- [3] Pan Fei, Fu Jingshun, Li Qiang, etc Exploration on the teaching system reform of C language programming under the background of new engineering[J], Education and Teaching Forum, 2020, (14):163-164
- [4] Wang Fang, Zhang Rui, Si Yujie, Research on the teaching reform of C language programming guided by improving the ability of program analysis and design [J]. Computer education, 2020(9): 168-173.