International Journal of Education and Social Development
ISSN: 3078-9931 (Print), ISSN: 3078-2287 (Online) | Vol. 5, No. 1, 2025

C Language Teaching Practice Based on Modular Design and File Operations

-- A Case Study of Classroom Roll Call System

Junwei Luo

Keyi College of Zhejiang Sci Tech Univ, Shaoxing 312000, China

Abstract: Aiming at the problems that students generally find grammar boring and theory divorced from practice in traditional C language teaching, this paper proposes to take the "classroom roll call system" as a comprehensive case, integrating two co re knowledge points of modular design and file operations into teaching practice. By guiding students to decompose the system into independent modules such as menu, roll call, student management and file operations, and using files to achieve data persistence, the students' programming interest, system design ability and problem-solving ability are effectively improved. Teaching practice shows that this method improves the teaching effect and provides a reference path for the training of applied talents.

Keywords: C language teaching, modular design, file operations, teaching case, classroom roll call system.

1. Introduction

C language is a core basic course for computer and related majors, but its abstract grammar and trivial knowledge points are likely to cause students to have a sense of difficulty. Especially in independent colleges, the goal of cultivating applied talents requires teaching to be close to reality. Traditional teaching often emphasizes grammar over practice, making it difficult for students to integrate concepts such as structures, pointers, and file operations [1].

Modular design can effectively manage program complexity, and file operations are the key to achieving data persistence [2-4]. The innovation of this paper lies in taking the "classroom roll call system" familiar to students as the carrier, and running the above knowledge points through a complete project, so that students can naturally master the comprehensive application of C language in the process of solving practical problems [5].

2. System Design and Module Implementation

2.1. System Requirement Analysis

As a software tool to assist teaching management, the classroom roll call system aims to improve the efficiency of classroom attendance and the convenience of management, and its functional requirements cover many aspects.

The random roll call function is one of the core functions of the system. It selects student names from the student list through a random algorithm to achieve fair and impartial classroom questioning or attendance checks.

The student list management function includes adding, deleting, and modifying student information. In actual teaching, the student list may change due to changes in majors, suspension of studies, resumption of studies, etc. Therefore, the system needs to have flexible student list management capabilities to ensure the accuracy and timeliness of data.

The file reading and writing function is the key to realizing persistent storage and reading of data. The system needs to be able to store important data such as student lists and roll call records into files for subsequent query and analysis. At the same time, when the system starts, it can read existing data from files to ensure the continuity of the system and the integrity of data. For example, during the attendance statistics at the end of the semester, teachers can quickly count each student's attendance by reading the roll call records in the file, providing a basis for the assessment of usual grades.

2.2. System Modular Design Idea

To realize the various functions of the classroom roll call system, a modular design method is adopted to divide the system into multiple functionally independent modules, each responsible for specific tasks.

The menu module is the entrance for user interaction with the system, providing an intuitive interface to display system function options, such as random roll call, student list management, file operations, etc. Users select the required functions through the menu module, and the system calls the corresponding modules for processing according to the user's selection. For example, when the user selects the "random roll call" option, the menu module transmits the selection information to the roll call module, triggering the random roll call operation.

The roll call module is responsible for implementing the core algorithm of random roll call, randomly selecting students from the student list, and returning the results to the menu module for display. In the implementation process, the randomness and repeatability of roll call need to be considered. To ensure randomness, the random number generation function rand in C language can be used, combined with the time seed srand to generate different random number sequences. For the repeatability problem, setting options can be provided according to actual needs, such as allowing or prohibiting repeated roll calls. When set to prohibit repeated roll calls, the roll call module needs to record the information of students who have been called to avoid repeated selection.

The student management module is mainly responsible for adding, deleting, and modifying student information. This module closely cooperates with the file operation module to

save data to files in a timely manner after student information is updated. When adding student information, it is necessary to verify the validity of the input information to ensure the accuracy and completeness of student ID, name and other information, such as checking whether the student ID is unique and whether the name is empty. If the input information does not meet the requirements, the user should be prompted to modify it in a timely manner. When deleting student information, it is necessary to confirm the user's operation intention to prevent accidental deletion of important data, and synchronously process relevant roll call records and other data to ensure data consistency.

The file operation module is responsible for implementing file reading and writing functions. When the system starts, this module reads the student list and roll call records from files, and loads the data into memory for use by other modules. During the operation of the system, when student information or roll call records change, the file operation module writes the latest data into files to achieve persistent data storage. In processing file operations, exceptions such as file opening failure and writing errors need to be considered. If the file fails to open, the user should be prompted to check the file path and permissions in a timely manner, and corresponding error handling measures should be provided, such as trying to reopen the file or create a new file.

Each functional module cooperates with each other to complete the various tasks of the classroom roll call system. The menu module serves as the control center of the system, coordinating the work of other modules; the roll call module, student management module and file operation module perform their respective duties, completing random roll call, student information management and data storage functions respectively.

2.3. Design of Each Functional Module

The design goal of the menu module is to provide users with a concise and intuitive operation interface to facilitate users to select system functions. A character interface is adopted, menu options are output through the printf function, and user input is obtained using the scanf function.

The roll call module mainly uses the random number generation function rand of C language to realize the random roll call function. To ensure that different random numbers are generated each time the program runs, the time function is used to obtain the current time as the random number seed, which is set through the srand function.

In the student management module, adding student information is done by the user entering the student ID and name, which are then stored in a structure array. When deleting student information, the corresponding student is found according to the student ID entered by the user and deleted from the array. When modifying student information, the student is found according to the student ID entered by the user, and then the student ID or name is modified according to the user's selection.

In the file operation module, the method of reading the student list from a file is to use the fopen function to open the file storing student information in read-only mode, and then use the fscanf function to read the student ID and name line by line from the file and store them in a structure array. The method of saving the list and roll call results to a file is to use the fopen function to open the file in write mode. If the file does not exist, a new file is created. Then, the fprintf function is used to write the student information and roll call results into the file. When saving roll call results, the results can be stored in an array first, and then written together when saving the file.

3. C Language Teaching Practice and Effect Evaluation

3.1. Teaching Practice Process

Introducing the classroom roll call system case into C language teaching, through carefully designed teaching steps, guiding students to gradually master relevant knowledge and skills. The specific teaching process is as follows:

Theoretical explanation: At the beginning of the course, systematically explain the basic concepts, principles and methods of C language modular design and file operations. Detail the advantages of modular design, such as improving code readability, maintainability and reusability. By comparing program examples with and without modular design, let students intuitively feel the optimization effect of modular design on program structure. For file operations, introduce basic operation functions such as file opening, closing, reading, and writing, such as fopen, fclose, fscanf, fprintf, etc. Explain the parameter meaning, return value and usage scenario of each function, and help students understand the basic process of file operations through simple file reading and writing examples.

Case analysis: In-depth analysis of the classroom roll call system case as the core teaching example. First, comprehensively introduce the functional requirements of the system, including random roll call, student list management, file reading and writing, interface interaction and other functions, so that students can understand the actual application scenarios and goals of the system. Then, detailedly explain the modular design idea of the system, showing how to divide the system into menu module, roll call module, student management module and file operation module, explaining the functions and responsibilities of each module, and how each module collaborates and transmits data.

Code demonstration: In class, teachers demonstrate the specific implementation process of each functional module of the classroom roll call system through actual code. During the demonstration, explain the role and logic of each line of code in detail, especially the key code parts involving modular design and file operations.

Student practice: After students understand the theoretical knowledge and are familiar with the case, arrange for them to conduct practical operations. Require students to independently complete the writing and debugging of the classroom roll call system according to what they have learned in class. During the practice process, students can refer to the code examples and explanations provided by teachers, but they need to think independently and solve the problems encountered. Teachers patrol and guide, answer students' questions in a timely manner, and help students overcome difficulties.

Group discussion and exchange: Organize students to conduct group discussions and exchanges, share experiences and problems in the practice process, and jointly explore solutions. Each group selects a representative to report the group discussion results and practical achievements, and other group members can put forward questions and suggestions. Through group discussions and exchanges, students can broaden their thinking, learn from others' programming experience and methods, and cultivate

teamwork and communication skills. During the discussion, teachers guide students to analyze and compare different solutions, understand their advantages and disadvantages, and thus select the optimal solution.

Summary and feedback: After students complete the practical tasks, teachers summarize and feedback the entire teaching process. Summarize the key knowledge points and programming skills of the classroom roll call system, and emphasize the importance of modular design and file operations in actual projects. Evaluate students' practical achievements, affirm their strengths and progress, point out existing problems and deficiencies, and put forward improvement suggestions. Encourage students to continue to improve the program after class to further consolidate the learned knowledge and skills. Through summary and feedback, help students have a clearer understanding of what they have learned and clarify their learning goals and directions.

3.2. Teaching Practice Results

Through the collection and analysis of teaching practice data, the following teaching practice results are obtained:

Student homework completion: In the homework related to the classroom roll call system, most students can correctly use modular design and file operation knowledge to complete the functions required by the homework. For example, in the homework of modifying system functions, about 70% of students can successfully implement function modifications, and the code structure is relatively clear with correct logic.

Exam scores: Compared with students using traditional teaching methods, students participating in this teaching practice have significantly improved their exam scores. The average score of students in this teaching practice is 3 points higher than that of students using traditional teaching methods, the excellent rate has increased by 7%, and the pass rate has increased by 10%. In the exam, students have a higher score rate in questions related to modular design and file operations, indicating that students have a good grasp of this part of the knowledge.

4. Research Conclusion Summary

After teaching practice, students can more proficiently use C language for program design. They have mastered modular design methods, learned to decompose complex problems into multiple independent functional modules, each realizing specific tasks, thereby reducing the difficulty of program development. Students' understanding of C language knowledge has also been further deepened. Through the specific case of the classroom roll call system, students are no longer limited to abstract grammar and concepts, but closely combine C language knowledge with practical applications. In the practice process, by continuously debugging and optimizing the code, students have a more solid grasp of basic C language knowledge such as syntax rules, data types, and operators, and can more accurately use this knowledge to solve practical problems.

In addition, teaching practice has also cultivated students' problem-solving ability and innovative thinking. In the process of implementing the classroom roll call system, students encountered various problems, such as the realization of random roll call algorithm, the uniqueness verification of student information, and the handling of file reading and writing errors. Through independent thinking, consulting materials and group discussions, students can analyze the essence of problems, put forward solutions, and verify the feasibility of the solutions through programming implementation. This practice process has exercised students' ability to solve practical problems and cultivated their independent thinking and innovative spirit. Some students have further expanded and optimized the functions of the classroom roll call system, such as adding attendance statistics functions and automatically generating attendance reports, showing students' innovative thinking and practical ability.

References

[1] Steering Committee for University Computer Courses of the Ministry of Education. Basic Requirements for University Computer Foundation Courses [M]. Beijing: Higher Education Press, 2016.

[2] Su Xiaohong, Chen Huipeng, et al. Practical University Course for C Language (5th Edition) [M]. Beijing: Electronic Industry Press, 2022.

[3] Kernighan B W, Ritchie D M. The C Programming Language (2nd Edition) [M]. Prentice Hall, 1988.

[4] Wang Shan, Sa Shixuan. Introduction to Database Systems (5th Edition) [M]. Beijing: Higher Education Press, 2014.

[5] Guo Yinzhang, Wang Lifang. Teaching Reform and Practice of C Language Programming Course Based on Project Task-Driven [J]. Computer Education, 2017, (02):41-44.