Reform and Research on Practical Teaching of "Data Structure"

Ruibin Xu[a], Rui Zhang[b]

[a]e-mail: 4771597@qq.com, * Corresponding author: [b]2374991185@qq.com

Quanzhou Institute of Information Engineering, Quanzhou,China

Abstract—"Data Structure" is a core course for computer majors in universities. The course teaching design is divided into two major stages: theoretical teaching and practical teaching. Among them, practical teaching is the teaching difficulty of this course and an indispensable teaching link. This article analyzes the current situation and existing problems of practical teaching in Data Structure, explores its causes, and studies the design and implementation plan of practical teaching reform in order to improve teaching quality.

Keywords-Data Structure; Practical Teaching; Reform in Education; Teaching Model

1. Introduction

"Data Structure" is a core course for computer majors in universities, such as software engineering, IoT engineering, network engineering, data science and big data, and cyberspace security. It is also a mandatory professional course for the computer science master's and postgraduate entrance exams. Therefore, it plays a crucial role in the entire undergraduate talent training program. By studying this course, students can become familiar with the rational organization of data, the selection of optimal structures, and the design of algorithms with better spatiotemporal efficiency involved in software design and development, thereby improving the overall quality of software design. However, these abilities are difficult to integrate solely through theoretical teaching. Only by transforming the difficult to understand and abstract concepts into visual programs through practical teaching can students consolidate and deepen their understanding of theoretical teaching content, cultivate their innovation and teamwork abilities, and improve software development skills.

2. Current Situation and Existing Problems in Practical Teaching of "Data Structure"

When studying this course, the theoretical content is mostly mastered by students through classroom teaching methods, such as initializing, searching, inserting, and deleting sequence tables. The teacher uses multimedia animation demonstration teaching to vividly impart knowledge points. At the same time, micro lesson videos of corresponding knowledge points have also been created in books. If you don't understand the lesson in class, you can also use your mobile phone WeChat to scan the code after class to watch and learn, reviewing the old

ITEI 2023, November 24-26, Zhengzhou, People's Republic of China

DOI 10.4108/eai.24-11-2023.2343675

and learning the new. However, practical teaching cannot be achieved solely through teacher instruction. The current situation and existing problems of practical teaching are briefly summarized as follows:

1) Less practical teaching hours

The proportion of practical teaching hours in the entire course teaching is relatively small. Generally, the total class hours for this course are 48, while the practical teaching hours are 16. There is one experiment every other week, which means there are only 8 experimental classes in the entire semester. This course is divided into 10 chapters, with one experiment arranged for each chapter, resulting in 10 experiments. The arranged practical teaching hours alone are not enough, and only some chapters of experiments can be performed, such as merging linear tables, traversing binary trees, minimum spanning tree algorithm for graphs, and construction of binary ordered trees.

2) Converting pseudo code to executable code is difficult

The code provided in books is mostly pseudocode, and it is difficult for students with poor foundation in C language to convert such pseudocode into executable code. Moreover, data structure programs do not have only a few or ten lines like C language programs. A executable data structure program includes preprocessing, structures, core algorithm code, sub functions, main functions, etc. These usually require dozens or even hundreds of lines of code, The duration of an experimental class is only 90 minutes, and most of the time is spent on code input, so there is no need to debug the core code carefully. How to successfully convert pseudo code into executable code? Only a few students with a solid foundation in C language can complete the experimental content, making it difficult to achieve the expected results of the experiment.

3) Lack of practical experimental guidance

So far, the practical teaching of this course lacks practical experimental guidance books. During class in the computer room, teachers usually design some validation experiments based on the textbook without improving the experimental guide book. Students only type ready-made code in the experimental class[1], without knowing the purpose and significance of the experiment. Even if there is an experimental guide book, only the experimental questions are given. This is too abstract, and students have no way to start and do not truly play a guiding role. It can be seen that in order to improve the effectiveness of practical teaching, it is crucial to write practical experimental guidance books. Only by applying the right medicine to the case can the disease be cured. For this purpose, I have compiled a school-based collection of experimental guidance, laying the foundation for the subsequent writing of experimental guidance books.

4) Simple evaluation method for practical teaching

The evaluation method of practical teaching is too simple, and in most cases, it is submitted in the form of experimental reports, which include the purpose, content, and results of the experiment. This method involves a large number of copying and pasting phenomena, and some students even submit it by simply changing the file name, without doing experiments themselves. Finally, the teacher gave a score based on the experimental report to complete the

evaluation. This overly simple experimental evaluation method seriously distorted the true meaning of practical teaching [2].

3. Design and Implementation Plan for the Reform of Data Structure Practical Teaching

The theoretical teaching and practical teaching of "Data Structure" are the two most important aspects of this course's teaching, which are interdependent. Just like the two wheels of a bicycle, theoretical teaching is the rear wheel, which is the foundation of the entire vehicle; Practical teaching is the front wheel, which guides the direction of the entire vehicle. As long as the direction is correct, practical teaching can visualize abstract and difficult to understand knowledge, which can cultivate students' ability to abstract data, innovate algorithm design, and standardize software design. Therefore, the reform and design of "Data Structure" practical teaching is very necessary and imperative. I have taken the course "Data Structure" multiple times, mainly based on the "13th Five Year Plan" national planning textbook for regular higher education undergraduate courses compiled by Professor Yan Weimin, combined with the reference of Professor Geng Guohua's high-quality course "Data Structure ", and proposed a reform design and implementation plan for the practical teaching of "Data Structure". Based on the OBE concept, from stimulating interest before class, releasing PTA experimental tasks (discovering problems) to strengthening basic or comprehensive experiments (solving problems) during class, conducting experimental reflection and process assessment and evaluation (promoting teaching through competitions) after class, in order to develop clear reform plans and implementation plans, as shown in Figure 1.

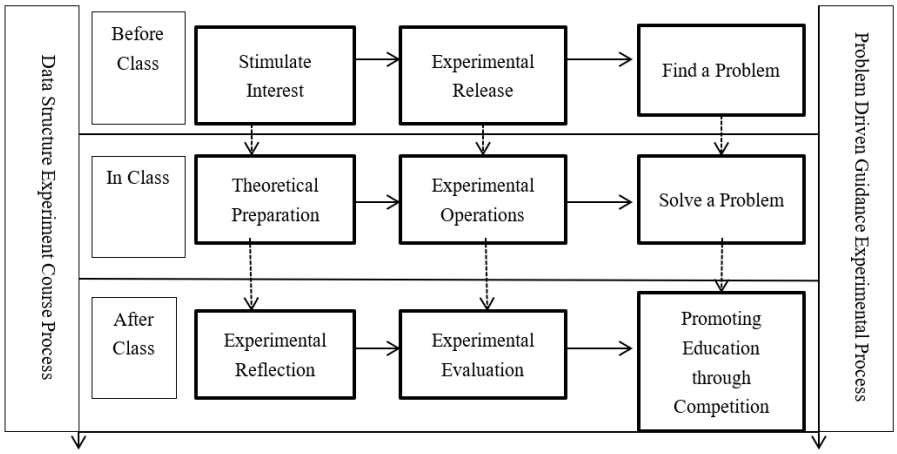

Figure 1. Implementation Plan for Data Structure Experimental Teaching Reform Design.

3.1. Stimulating interest and discovering problems

Interest is the driving force behind all learning[3]. Without interest, all learning will inevitably become dull and uninteresting. In order to stimulate students' interest in learning, teachers can adopt the OBE concept, focus on combining with reality, find some cases that are closely

related to real life, and encourage students to resonate in their thoughts. They also encourage students to discover problems and conduct experiments with problems. Such experiments are both fun and have a sense of achievement after completion. For example, when explaining the data structure of the graph, taking the shortest path problem in campus navigation as an example, taking real buildings on campus such as dormitories, teaching buildings, experimental buildings, libraries, canteens, and gyms as vertices, using adjacency matrix or inverse adjacency matrix methods to store important buildings on campus, and using Dijkstra algorithm to solve the shortest path between different vertices. For example, when explaining the queue data structure, the dance partner problem can be introduced. Assuming that at a weekend dance, when men and women enter the dance hall, they each line up. At the beginning of the dance, one person from each of the men's and women's teams is selected as the dance partner, and the man or woman who enters the team first should be selected as the dance partner. This problem has a typical first in, first out characteristic and a data structure that conforms to queue characteristics. These real-life examples are relatively easy for students to understand and accept. On the day before each experimental class, publish the experimental content through the PTA platform, guiding students to preview and prepare first.

3.2. Pay attention to the connection between the preceding and subsequent courses, and clarify the significance of practical teaching

The process of cultivating computer professionals attaches great importance to the main line setting, which is the connection between the leading courses and subsequent courses, just like a linear table, with direct predecessor nodes and direct successor nodes. As the leading course of the core course "Data Structure" in computer science, it is "Advanced Language Programming (C Language)". The theoretical knowledge points such as functions, pointers, structural entities, and the proficient application of the experimental platform VC++in this course are all important connections for the study of "Data Structure". Although students have studied this part of the content, many students are not yet familiar with its practical application, especially the application and debugging of experimental platforms. Therefore, before the start of the course "Data Structure", it is necessary to spend two or three times reviewing the key and difficult points of C language, familiarizing and applying the VC++experimental platform, such as the keywords warning and error appearing in debugging, what these two prompts mean, and how to modify the debugging, so that students can review the old and learn new things, laying a solid foundation for the learning of subsequent chapters.

The purpose of practical teaching of "Data Structure" is not only to design an algorithm and write a few lines of code, but more importantly, to cultivate students' rigorous logical thinking ability to support various knowledge points in theoretical teaching. They are good at using teamwork to analyze and solve problems, and ultimately master all knowledge points of this course, filling the gap for future postgraduate entrance exams.

3.3. Strengthen validation experiments and write experimental guidance books that are suitable for practical learning situations

The knowledge points in each chapter of 'Data Structure' are relatively scattered and have the characteristics of being relatively abstract and difficult to master. A validation experiment is an algorithm condensed from the knowledge points in each chapter, which is a small piece of pseudocode that must be converted into an executable program to run on the VC++platform.

Validation experiments are designed to help students better understand the knowledge points in each chapter and master the practical application of algorithms. They are content that each student is required to complete independently in practical teaching. However, for students in general application-oriented undergraduate colleges, their programming foundation is generally poor, and they lack the perseverance and perseverance to self-study. In order to improve the effectiveness of practical teaching, it is particularly important to write experimental guidance books that are suitable for practical learning situations. Clarify the experimental purpose, experimental platform, experimental principles, and experimental content of each validation experiment in the experimental guide book. A good experimental guide book can provide students with a twice the result with half the effort to successfully complete each experiment.

In recent years, there have been many theoretical textbooks for data structure courses, some in the C language version and some in the Python version, but they are rarely equipped with experimental guidance books. Even if there are experimental guidance books, there are only experimental problem sets, and students have no way to start when conducting experiments; Either there is a complete program code that students can simply copy and run when conducting experiments, without having to think too much. Such experimental guidance books are not practical and cannot improve the understanding and mastery of various knowledge points in the textbook through experiments[4]. A good experimental guide book should first cover the important knowledge points of the textbook, without having to cover every aspect. Similar knowledge can allow students to imitate and debug on their own. For example, in the traversal of binary trees, there are pre order traversal, mid order traversal, and post order traversal. Therefore, as long as the purpose, principle, and experimental content of pre order traversal are clearly defined in the experimental guide book, students can use the recursive methods learned in theoretical courses to complete the experiment of pre order traversal, that is, to run and debug the program on a computer. After class, the experiments of middle order traversal and post order traversal will be published to students using PTA, requiring them to complete them within the specified time. This can serve as an extracurricular assignment to supervise students' daily learning process and strengthen their mastery of the important knowledge point of binary tree traversal. Secondly, the experimental guide book should provide corresponding framework source code for each validation experiment, such as preprocessing commands, structured programs, partial subfunctions, and main functions, and hide the code of the core algorithm. Students should be required to complete the thinking and writing of the core code according to the experimental principles and content of the experimental guide book within the specified time of the experiment, and then debug it. During this time, teachers or classmates can discuss it with each other, Finalize the corresponding experimental report. After years of practical teaching experience, I have developed a realistic experimental guidance school-based collection that has been used by students majoring in software engineering at levels 20, 21, and 22 of our school. Finally, we have formed a course teaching team to further revise and improve the school textbook for this experimental guidance. We will produce an experimental guidance book in the future, hoping to help improve the quality of practical teaching.

3.4. Using projects as a carrier to select design based experiments and expand practical abilities

Validation experiments mainly aim to design and validate corresponding algorithms for the knowledge points required in each chapter of the "Data Structure" course, with the aim of enabling students to better grasp the most basic knowledge points and basic concepts of data structure, while cultivating students to independently complete the programming of simple algorithms. On the basis of validation, a project-based approach was proposed to select several design based experimental projects. Firstly, it can increase students' interest in hands-on experiments, secondly, it can expand students' practical abilities, and thirdly, it can cultivate students' teamwork spirit and enhance their teamwork ability for each student who will participate in various competitions[5]. The design based experimental projects are shown in Table 1. Students can work in groups of several and choose to do a few of the projects. After completion, they will be handed over to the teacher, which can serve as bonus points for the process assessment.

Table 1. Design based experimental projects

Serial Number	Project Name	Experimental Focus
1	Integer arithmetic unit	linear list
2	Parking management system	queue
3	Nucleic acid detection of	Pattern matching of strings
4	COVID-19 Teletext compiler	Huffman tree
5	Campus Tour	Picture
6	address book	Find and Sort

3.5. Assessment and Evaluation of Practical Teaching Process

Practical teaching runs through the entire semester, and effective supervision, assessment and evaluation measures should be taken for each experiment, with the aim of giving each student a corresponding sense of urgency and time awareness. In order to improve the achievement of each experiment's goals, the assessment method has been reformed, which is also in line with the concept of engineering certification. The proportion of process assessment has been increased, while the proportion of result assessment has been reduced, with process assessment accounting for 70% and result assessment accounting for 30%, respectively. Among them, the process assessment includes five aspects: online learning completion, classroom attendance, interactive discussion, computer debugging process, and program operation results. Teachers can observe and supervise the online status of each student in real-time on the teacher's computer through the "Lingbo" multimedia software, and register and record the process evaluation and assessment scores. The result assessment includes the experimental reports and design results submitted by students, and scores are evaluated based on the completeness and accuracy of the experimental reports. The total score of the comprehensive process assessment (70%) and the result assessment (30%) will be used as the final score for an experimental class. This comprehensive and multi-angle assessment method achieves comprehensive assessment and evaluation of all aspects of practical teaching, and can objectively reflect the practical effects of students in each experimental class. Through the questionnaire survey and the final student evaluation results, it can be seen that the overall

satisfaction of students with the "Data Structure" course has increased from 70% to the current 90%.

3.6. Promoting Education through Competition and Motivating Achievement Transformation

In recent years, we have successively set up a number of professional associations and interest groups. Every day, we use the spare time or weekend time to concentrate in the computer room of the school, and the instructors give training and guidance in turn. The main events are the "Blue Bridge Cup" National Software and Information Technology Professionals Competition, China "Internet plus" Undergraduate Innovation and Entrepreneurship Competition, and China Undergraduate Service Outsourcing Innovation and Entrepreneurship Competition. The year before last, students from our college won one first prize and two second prizes in the "Blue Bridge Cup" provincial competition; Last year, I participated in the "Blue Bridge Cup" and won two second prizes and three outstanding awards in the national competition. Allowing students to participate in such competitions enables them to compete in programming skills with students from universities such as Peking University and Xiamen University. Students not only gain insight and broaden their horizons, but also have more motivation and confidence in their own learning.

4.Conclusion

Data Structure is a core course in the talent training program for computer majors in undergraduate education, and it is essential for improving logical thinking and algorithm design optimization abilities in software engineering. The theoretical teaching of this course can be carried out through various methods such as classroom lectures, teaching videos, MOOCO courseware, etc. However, its inherent complexity and abstraction make it difficult for students to master important knowledge points. Only through practical teaching can these difficulties be programmed and visualized, transformed from abstract to concrete, and from complex to simple [6]. This article combines the author and teaching team's years of teaching experience in "Data Structure" and proposes effective reform measures. It summarizes practical teaching experience from stimulating interest, clarifying teaching significance, designing validation based experiments, selecting design based experiments, and reforming evaluation and assessment. It has been successfully applied to students majoring in software engineering at our school's 2020, 2021, and 2022 levels. By comparing the academic performance of students in these three grades in the course 'Data Structure', it is evident that the failure rate has decreased and the excellence rate has slightly increased. In short, theoretical teaching and practical teaching are the two key points in the teaching of the "Data Structure" course, while practical teaching is a weak point and also a teaching bottleneck, which is worth every teacher's serious thinking and continuous reform and innovation[7]. How to break through this bottleneck and better integrate it with theoretical teaching, thereby improving the teaching quality of the "Data Structure" course, is a teaching reform topic that every teacher and teaching team constantly strive to research.

References

[1] Yan Weimin, Li Dongmei, Wu Weimin. Data Structure (C Language Edition | 2nd Edition) [M]. Beijing: People's Post and Telecommunications Press, 2021

[2] Yejee L,Hayoung B. L-rCBF: Learning-Based Key–Value Data Structure for Dynamic Data Processing[J]. Applied Sciences,2023, 175: 45–51.

[3] Qu Weilan, Li Xiaohong, Yang Xiaobo, Luo Jiawei, Xiao Zhu. Data structure experimental teaching reform and practice based on "six dimensional integrated innovative teaching closed-loop" [J]. Computer Education, 2021 (4): 77-81

[4] Xiang Zhihua, Lai Xiaoping. Construction of a practical teaching system for "data structures and algorithms" guided by innovation ability [J]. Wireless Internet Technology, 2020, 17 (1): 83-85

[5] Yin Fengmei, Shi Junhua. Exploration of Online Teaching Mode for Data Structure Courses Based on OBE [J]. Journal of Changchun University, 2021, 31 (04): 96-100

[6] Zhang Liling. Innovative Design and Practice of Data Structure Curriculum Teaching with Ability Cultivation as the Core [J]. Journal of Hubei Second Normal University, 2023,40 (02): 81-85

[7] Ogayar-Anguita Carlos J. Nested spatial data structures for optimal indexing of LiDAR data[J]. ISPRS Journal of Photogrammetry and Remote Sensing,2023, 261: 65–72.