Exploration of Teaching "Data Structures" in Applied Undergraduate Programs: A Case Study of Internet of Things Engineering

Xiaoqiang Luo[1], Hairong Zhu[1,*], Yilan Zeng[2]
[1]Guangxi Normal University of Science and Technology, Laibin, Guangxi, China
[2]The Second Junior High School of Nanmu Town, Guigang City, Guangxi, China
**Corresponding Author.*

Abstract: In order to further improve the teaching effectiveness of the "Data Structures" course and ensure that this historically challenging subject is taught well, this paper addresses five key issues pertaining to prerequisite knowledge, textbooks, course design, teaching methods, and pedagogical principles, based on the author's observations in their own university teaching. The paper proposes six strategies tailored to the characteristics of the discipline, including the selection of appropriate textbooks, emphasis on practical components, optimization of teaching content, innovative teaching methods, enhanced interaction and personalized guidance, and the integration of outcome-oriented educational concepts. Through observations over four academic years, these approaches have been found to significantly improve the teaching effectiveness of the course.

Keywords: Data Structures; Teaching Effectiveness; Prerequisite Knowledge; Textbooks; Course Design; Teaching Methods

1. Introduction

In the context of daily teaching of "Data Structures" at applied undergraduate institutions, students commonly encounter the perception that the subject is excessively abstract and extremely challenging to learn. Drawing on the author's teaching experience over four academic years, this paper explores reformative strategies for the teaching of "Data Structures" from the perspectives of textbook selection, course content, and teaching methods. The aim is to facilitate more efficient learning and mastery of the course among students in relevant disciplines [1].

The "Data Structures" course constitutes a core subject within computer-related disciplines such as Internet of Things Engineering, Computer Science and Technology, and Software Engineering. It falls under the category of fundamental courses in the field, making it of paramount importance to ensure that students acquire a solid understanding of its content [2][3].

However, the level of proficiency expected in "Data Structures" may vary across different tiers of educational institutions. For research-oriented universities, the emphasis should be placed on nurturing students' innovative thinking [4]. Conversely, for applied undergraduate institutions, the focus should primarily lie in comprehending classical data structures and algorithms, subsequently mastering their practical application to achieve the learning objectives [5]. Therefore, based on the principle that applied disciplines should prioritize practical application, this paper delves into the discussion of the challenges present in the teaching process and proposes reformative strategies for the course [6].

2. Problems in the Teaching Process of the Course

2.1 Inadequate Mastery of Prerequisite Knowledge

As the saying goes, "a building with a weak foundation will collapse in an earthquake." This phrase aptly describes the situation of students in our major each academic year. Many students often struggle to pass the final exam of "C Language Program Design," which is a prerequisite course for "Data Structure." Based on the past data and experience, it is evident that many students have difficulty mastering the fundamentals, as they are unable to grasp the basic concepts of “C Language Program Design.” Consequently, when it comes to studying “Data Structure,” students often exhibit fear, aversion to learning, engage in distractions such as playing with their phones during class, and sometimes even skip classes.

2.2 Unsuitable Textbook Selection

In the selection of textbooks, many applied universities still choose the “Data Structure” textbook authored by Professor Yan Weimin from Tsinghua University. This textbook has won numerous awards for excellence due to its comprehensive theoretical foundation, strong logical and abstract reasoning, and its effectiveness in cultivating logical thinking and data abstraction capabilities in computer science students [7]. However, since its publication, the substantive content has remained largely unchanged. Given that our university primarily focuses on cultivating applied talents, this textbook strictly uses algorithm design language to describe algorithms. As our students are not proficient in the basics of “C Language Program Design,” it becomes challenging for them to convert the algorithms in the textbook into practical programs. This situation does not facilitate the improvement of students' practical abilities.

2.3 Unreasonable Course Arrangement

Every year, a considerable number of students at our university retake the “Data Structure” course. For non-Internet of Things engineering students who retake the course, there are instances where the course content does not align with what they have previously studied. For example, some majors heavily focus on learning the Java language and do not offer a course in C language. Consequently, these students have not learned the content related to pointers and structures in “C Language Program Design.” When these students retake the “Data Structure” course in the Internet of Things engineering program, they struggle to comprehend the content related to pointers, thus falling behind the teachers' pace. Therefore, the mixed-class learning method for different majors does not cater to their individual learning needs and does not facilitate the acquisition of professional knowledge.

2.4 Monotonous Teaching Methods and Lack of Interaction

The content of the “Data Structure” course is abstract and relatively difficult to understand. It is also inherently theoretical and often perceived as dull and uninteresting. Some teachers, in their rush to cover the curriculum, solely deliver lectures without incorporating practical exercises and fail to make the content engaging enough to capture students' attention. At times, students may struggle to comprehend the material and can only resort to diligently taking notes and listening attentively. Such teaching methods may lead students to become drowsy within half a class, thereby failing to achieve effective teaching outcomes.

Furthermore, the lack of opportunities for student participation and interaction results in passive knowledge acquisition, with a dearth of interactive communication. Students rarely have the chance to ask questions, engage in discussions, or share their own interpretations and viewpoints. This contributes to a monotonous classroom atmosphere and hampers the achievement of the desired learning outcomes.

2.5 Lack of Engineering Application Philosophy

In most cases, many “Data Structure” course instructors solely focus on explaining theoretical knowledge points without incorporating practical applications into the course structure. This results in students not only being unable to fully grasp the knowledge but also lacking opportunities to apply their acquired knowledge. Consequently, they are unable to achieve the goal of applying what they have learned. For certain knowledge points, students can initially learn how to use them and then deepen their understanding through application.

Overall, these issues in the teaching process of the course “Data Structure” necessitate a reevaluation and restructuring of the curriculum, teaching methods, and course materials to better meet the needs of the students and enhance the effectiveness of the learning process.

3. Curriculum Reform Ideas

3.1 Selecting Textbooks Based on Professional Characteristics

In my practical teaching experience, I have used the book “Data Structures” by Zhou Guihong, published by Nankai University Press. There are several reasons why I chose this textbook: First, as we are in the field of Internet of Things engineering, students will need to engage in hardware programming in the future, which naturally involves programming with pointers. Therefore, a version that does not describe the concepts in C language cannot be adopted. Second, this textbook provides simple and complete C language example code, allowing students to directly run the code based on their interests and practical applications.

Considering the non-major students who are retaking the course, if they have studied the Java version of “Data Structures,” they may not be familiar with the concepts of pointers and structures in C language. As a result, they may struggle to keep up with the learning progress. Therefore, I suggest that these students retake the course in their own major's class instead of the C language version.

3.2 Emphasizing Practical Components

The course “Data Structures” is divided into nine chapters: Introduction, Linear Lists, Stacks and Queues, Strings and Arrays, Trees, Graphs, Sorting, Searching, and Files. For each chapter that I teach, there are corresponding exercises and experiments. The introduction and strings/arrays chapters mainly have exercise assignments, while the other chapters have one or two experiments associated with them. For example, the linear lists chapter corresponds to two experiments: “Application of Sequential Lists” and “Application of Linked Lists.” These experiments highlight practical applications that align with the professional training direction [8][9]. During the lectures, in addition to teaching theoretical knowledge, I allocate appropriate time to guide students in implementing the experiments. If time is insufficient, students can complete the remaining content individually or in groups. The assessment method not only checks the completion status but also flexibly tests the functionality of the implemented features, reducing the occurrence of plagiarism and superficial completion.

3.3 Optimization of Teaching Content

In the teaching process, we often want to explain every knowledge point in the textbook to the students. However, due to limited class hours, we need to make appropriate choices. Some content can be briefly introduced without in-depth lectures, providing only simple background information. For example, after introducing the adjacency list and adjacency matrix storage structures for graphs, we can briefly mention the linked representation and multiple linked representation and move on. Similarly, since file handling has been introduced in the C language, we only demonstrate and briefly introduce its application.

3.4 Innovative Teaching Methods

In terms of teaching methods, I have abandoned traditional single methods and integrated lecture-based teaching, group work, project-based learning, etc. Additionally, I have adopted a combination of offline and online teaching methods. Considering that students have varying levels of mastery of foundational knowledge, I conduct brief tests to assess their understanding. Based on the test results and voluntary participation, I form study groups for mutual assistance. Each knowledge point is paired with an application experiment, and if time permits, I randomly select groups to present the knowledge points on the podium. Regular assignments are mainly completed online using the “Chaoxing” platform, and detailed answers to the exercise questions are provided to improve students' learning efficiency [10].

3.5 Enhanced Interaction and Personalized Guidance

I incorporate interactive elements in the classroom, such as group discussions and question-and-answer sessions, to promote communication and cooperation among students. In the data structure experiments, I use group discussions, allowing students to form their own groups freely. I also individually assess the results of the experiments, effectively avoiding situations where group members do not collaborate. This approach not only improves students' learning outcomes but also cultivates their teamwork and communication skills.

To provide personalized guidance and support, I employ different teaching methods and strategies based on each student's learning level and needs. During the tutoring process, I primarily provide one-on-one guidance online or offline, as well as extracurricular tutoring groups, to help students resolve learning issues and improve their learning efficiency.

3.6 Integration of Outcome-Oriented Education Philosophy

The key to integrating the outcome-oriented education philosophy is to closely link students' learning experiences with practical applications, cultivating their engineering practical abilities [11]. By setting goals and evaluation criteria, introducing engineering case studies, fostering problem-solving abilities and teamwork, and paying attention to student feedback, we can adjust teaching strategies, stimulate autonomous learning, keep up with industry development, and continuously update the curriculum. By incorporating these curriculum reform ideas, the “Data Structures and Algorithms” course will align more closely with the outcome-oriented education philosophy and cultivate more applied talents with engineering practical abilities.

4. Conclusion

The course “Data Structures” is a core foundational course for the Internet of Things Engineering major, playing a crucial role throughout the four years of university education and having a significant impact on the entire academic career. In view of the highly abstract and logically challenging nature of the course content as reflected by students, the author, based on personal experience, has analyzed the main issues found in the current teaching process and put forward reform suggestions for the “Data Structures” course. It is hoped that these suggestions will be helpful for curriculum teaching at applied undergraduate institutions, especially for teachers who are new to university teaching.

The “Data Structures” course is essential for shaping students' understanding of fundamental programming concepts and algorithmic thinking, which are indispensable for their future careers in the field of Internet of Things engineering. However, the complexity and abstract nature of the course content often pose challenges for students, leading to difficulties in learning and comprehension. Therefore, it is crucial to reform the teaching approach and methods to better facilitate student learning and understanding.

The reform suggestions presented in this proposal aim to address the following key areas:

1. Textbook Selection: Choosing textbooks that align with the professional characteristics of the major and cater to the practical needs of the students, ensuring relevance and applicability.
2. Emphasizing Practical Components: Integrating practical components such as exercises and experiments to reinforce theoretical knowledge and promote hands-on learning experiences.
3. Optimization of Teaching Content: Streamlining the teaching content to focus on essential knowledge points and omitting unnecessary details to improve learning efficiency.
4. Innovative Teaching Methods: Implementing diverse teaching methods including group work, project-based learning, and a combination of offline and online teaching to enhance engagement and understanding.
5. Enhanced Interaction and Personalized Guidance: Encouraging student interaction through group discussions and individualized guidance to cater to varying learning needs and levels.
6. Integration of Outcome-Oriented Education Philosophy: Aligning the course with an outcome-oriented education philosophy by setting clear goals, introducing engineering case studies, and fostering problem-solving abilities.

By implementing these reform suggestions, the “Data Structures” course can be transformed into a more engaging, relevant, and effective learning experience for students. It is essential to continuously evaluate and adapt the teaching strategies based on student feedback and industry developments to ensure the course remains aligned with the evolving needs of the field.

Ultimately, the goal of these reform proposals is to cultivate a new generation of applied talents with strong engineering practical abilities, equipped to meet the demands of the rapidly advancing field of Internet of Things engineering. Through collaborative efforts and a commitment to continuous improvement, the “Data Structures” course can serve as a cornerstone in preparing students for successful and impactful careers in the field of technology and engineering.

Acknowledgments:
This work was supported by Project for Enhancing Young and Middle-aged Teacher's Research Basis Ability in Colleges of Guangxi (Gui Ke Education Research [2021] No.1) 2021KY0853, 2021KY0859.

References:

[1] Wang Yun. Exploration and Reflection on the Teaching of “Data Structures” Course in Application-oriented Undergraduate Programs. Science and Innovation, 2022(05):33-35+42.
[2] She Xingxing. Exploration of Teaching Models for Data Structures and Algorithms Courses in Application-oriented Undergraduate Software Engineering Programs. Higher Education Journal, 2024, 10(S2):103-106.
[3] Chen Yongqiang. Practical Reform of Flipped Classroom Model Based on MOOC for Application-oriented Undergraduate Computer Science Programs. Education and Teaching Forum, 2020, (50):165-166.
[4] Xu Hang, Xie Ying, Yan Tao, et al. Reform and Exploration of Data Structures Course Based on Blended Teaching. Higher Education Journal, 2023, 9(16):148-151+157.
[5] Sun Yue, Jiang Jing, Su Limin. Skillful Use of Cases in Data Structures Course Design Teaching. Science and Technology Wind, 2024, (04):97-99.
[6] Li Zhongwen, Karen H. Jin, Zhang Zhiqiang. Current Status and Experiences of Teaching Reforms in “Data Structures” Courses in Universities. Information Systems Engineering, 2022, (10):161-164.
[7] Gao Yuan, Fu Li, Huang Chunguang. Exploration of Reforming Online and Offline Blended Teaching Models under the Background of Broad Enrollment Categories: Taking “Algorithms and Data Structures” Course as an Example. Heilongjiang Education (Theory and Practice), 2023, (07):80-82.
[8] Wang Yan, Pan Xiaoying, Luo Jiaqi, et al. Exploration of Online Teaching Reform in Data Structures Courses Empowered by Education. Science, Education and Culture Digest, 2022, (21):65-68.
[9] Liu Yuxiu, Li Yi. Student-centered Practical Teaching Exploration of “Data Structures and Algorithms”. Computer Era, 2023, (09):155-158.
[10] Yan Hong, Bai Hexiang, Li Qingyun. Practical Teaching Exploration of Data Structures and Algorithms for New Engineering Disciplines. Computer Education, 2023, (09):74-77.
[11] Yu Zhen. Research on Teaching Reform of Data Structures and Algorithms Course Based on OBE Concept. Science and Technology Wind, 2023, (19):123-125.

