Journal of Education and Educational Research
ISSN: 2957-9465 | Vol. 7, No. 3, 2024

Methods and Practices in C Language Teaching

Liangliang Gu

School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai, 200093 China

Abstract: The C language holds significance as a pivotal programming language with wide-ranging applications. This paper delves into the methodologies and practical approaches employed in teaching C language as a programming tool. Through an examination of student learning requirements, comparison of various teaching methodologies, and analysis of practical cases, this paper delineates the pivotal factors contributing to success in C language instruction. By devising tailored pedagogical plans and integrating programming exercises into real-world projects, student proficiency in programming can be enhanced, resulting in favorable educational outcomes.

Keywords: C Language; Teaching Methods; Teaching Practices.

1. Introduction

Over the past two decades, the study of C language has been an obligatory component for students pursuing majors in science and engineering within Chinese higher education institutions. With time, the educational resources for this course have proliferated significantly, encompassing textbooks, online materials, programming exercises, community forums, and software tools. Furthermore, instructional strategies have undergone continual refinement, progressing from initial theoretical exposition to the amalgamation of theoretical knowledge with practical application, and subsequently advancing towards collaborative project development within teams. There exists a growing emphasis on nurturing and honing students' practical proficiencies. Presented below are summarized insights into prevalent teaching methodologies and practices currently observed in the field.

2. Teaching Methods for C Language

The pedagogical approaches to teaching C language exhibit considerable diversity, primarily categorized into two main classifications. The first classification entails traditional theoretical instruction coupled with collaborative learning techniques, focusing on elucidating syntax and semantics, alongside rigorous exercise regimens. Conversely, the second classification comprises project-oriented instruction integrated with tailored learning methodologies. This methodology entails elucidating conceptual frameworks and definitions through case studies, while augmenting comprehension through practical projects.

There exists a discernible variance in textbook selection and assessment criteria between these two instructional methodologies. The former approach commonly relies on seminal works such as "C Primer Plus" and "The C Programming Language," renowned for their thorough and comprehensive theoretical expositions. Conversely, the latter method frequently employs contemporary textbooks rich in case studies, such as "Programming Methodology and Techniques - C Language" and "C Language Programming Case Tutorial." These texts feature detailed case studies in each chapter, elucidating fundamental concepts and their practical applications, thereby facilitating students' comprehension of abstract principles.

Undoubtedly, project-driven teaching methodologies have become increasingly apt for contemporary educational contexts. By immersing students in practical applications, this approach facilitates experiential learning. The design of challenging projects not only piques students' interest but also nurtures their sense of initiative. Furthermore, by encouraging students to apply their knowledge to address real-world problems within these projects, critical thinking and problem-solving skills are honed, enriching the learning experience.

Through a comprehensive understanding of students' learning styles and proficiency levels, instructors can implement differentiated teaching strategies. These strategies encompass offering exercises and projects spanning varying difficulty levels to accommodate diverse student requirements. Leveraging online learning platforms fosters opportunities for independent learning and facilitates the identification of students' personalized learning preferences. The design of timely and constructive feedback mechanisms aids students in comprehending errors and rectifying them promptly, thereby enhancing learning outcomes.

Moreover, activities such as group discussions and programming challenges foster collaboration among students, enriching the educational experience. Encouraging students to pose questions acknowledges that problem-solving is an integral facet of the learning process, thereby fostering a dynamic and engaged learning environment.

3. Teaching Resources and Tools for C Language

In addition to the selection of diverse textbooks and practical application cases, a plethora of online resources are available for C language instruction, including video tutorials, online documentation, and programming examples. Integrating open-source projects and community resources enables students to engage in real-world projects, thereby fostering continuous improvement of teaching resources. The utilization of online learning platforms or automated testing tools expedites the feedback process, facilitating peer code reviews and fostering collaborative and interactive learning environments.

Hence, conventional resources such as C language standard documents, compilers (e.g., GCC/Visual Studio), debugging tools (e.g., GDB/DDD), and integrated development

environments (e.g., Code Blocks/Dev C++) complemented by online platforms like GitHub, LeetCode, and CSDN open-source project platforms serve to significantly enrich teaching resources. This integration effectively enhances students' learning depth and breadth by providing a comprehensive array of tools and platforms for theoretical understanding and practical application in C language instruction.

Furthermore, as teaching hardware facilities undergo continuous upgrades, interactive whiteboards, comprehensive online learning platforms, and online programming tools can be leveraged to enhance classroom interactivity. These resources facilitate real-time checking and feedback, aligning with the "learning while doing" pedagogical approach.

4. Practical Implementation in C Language Teaching

The integration of practical case studies in teaching serves to exemplify the diverse applications of C language across various domains. This approach aids students in developing a profound understanding of programming principles through the analysis of real-world examples and immersion in authentic programming environments. Furthermore, practical case studies foster the cultivation of teamwork and communication skills among students.

Moreover, the implementation of practical case studies allows for the holistic evaluation of students' performance from multiple perspectives. Personalized evaluations and recommendations can thus be provided to address individual learning needs effectively. Additionally, regular feedback sessions are conducted with students to solicit their needs and opinions, ensuring that the teaching methods remain responsive to evolving student requirements.

By adopting a comprehensive approach that incorporates these methods, educators can effectively cater to the diverse learning needs of students. Furthermore, this approach serves to enhance students' engagement in learning and their practical programming capabilities.

To cater to varying learning objectives at different stages of the course, it is imperative to design diverse types and difficulty levels of case studies. For instance, at the onset of the course, engaging students' interest in learning C language can be achieved through game development projects. Students can be tasked with developing simple games utilizing C language, such as Tic-Tac-Toe, Snake, or Tetris. These interactive and practical applications effectively stimulate students' enthusiasm for learning.

In the middle phase of the course, once students have acquired proficiency in basic syntax and statement structures, the introduction of embedded systems and Internet of Things (IoT) applications is pertinent. Examples include temperature monitors or smart lighting systems commonly encountered in daily life. Such cases necessitate both hardware and software knowledge, integrating C language with embedded hardware to enable students to experience the intricacies of hardware-software interaction. This reinforces the realization that C language serves as a potent and versatile tool in practical applications.

Towards the culmination of the course, upon students' grasp of modular programming concepts, case studies can focus on designing and implementing a straightforward student performance management system. This entails functionalities such as student information management, initial grade entry, GPA calculation, and query capabilities. Project requirements and functionalities, alongside potential extensions, are elucidated to underscore the project's practical applicability. Students are guided through the project's design process, including the delineation of functional modules and the design of data structures. This aids in formulating project plans and clarifying tasks and deadlines for each phase. Through this comprehensive case study, students gain insights into the significance of modular programming concepts and the value of collaborative teamwork.

Case-based teaching permeates the entirety of the teaching and learning journey, characterized by continual experimentation and iteration to accommodate the swiftly evolving technological and educational landscape.

5. Conclusion

In conclusion, the teaching methodologies and practices within C language instruction underscore the significance of case studies and innovative pedagogical approaches in augmenting students' theoretical comprehension and programming proficiencies. The adoption of diverse teaching methodologies in C language instruction, coupled with the utilization of online resources and open-source tools, is imperative to address the personalized learning requirements of diverse student cohorts. This comprehensive approach fosters the development of students' aptitude in applying acquired knowledge and skills within collaborative settings, thereby enhancing their capabilities in teamwork and integration.

References

[1] Xie, L., (2004). Programming Design and Applications. Beijing: Metallurgical Industry Press.

[2] Ke, X., (2006). A Brief Discussion on the Teaching of C Language Programming Courses in Universities. Journal of Anyang Institute of Technology.

[3] Li, N., (2008). Discussion on the Reform of C Language Programming Course Teaching. Computer Knowledge and Technology.

[4] Wang, B., (2009). Discussion on Teaching Methods of C Language Programming Courses. Computer Education.

[5] Cheng, Z., Liu, G., & Li, R., (2009). Discussion on Teaching of C Language Programming Experiment Course for Non-Computer Majors. Computer Knowledge and Technology.