

Exploration of Multidimensional Teaching Method Around Assembly Language Programming Course

Ying Liu^{1,a,*} and Fuxiang Gao^{1,b}

¹College of Computer Science and Engineering, Northeastern University, Shenyang, Liaoning, China

^aliuying@cse.neu.edu.cn, ^bgaofuxiang@mail.neu.edu.cn

*corresponding author

ABSTRACT

Aiming at the problems existing in the teaching content and process of assembly language program designed for computer majors in colleges and universities, the key problems such as course motivations, teaching methods, practice teaching and course assessment reform were discussed in this paper, and the reform ideas and implementation plans were put forward.

Keywords: assembly language programming, multidimensional teaching method, teaching reform ideas

1. INTRODUCTION

Assembly Language Programming is a basic course for students majoring in computer science and technology. It is not only a necessary precursor course for other courses such as computer composition principle, interface technology and operating system, but also plays an important role in training students to master programming technology, proficiency in computer operation and program debugging technology. However, according to the feedback results of long-term teaching effect, most students were afraid of this course, found that it was difficult to grasp the relevant knowledge points. There were two reasons for this: (1) the compatibility of assembly language and hardware was high, students should learn some required hardware courses in advance before learning this language, and the language instruction set was large, the language structure was relatively complex, which required students to have a certain programming ability; (2) when learning assembly language programming the structure of the program was not obvious, and the debugging results on the computer were not as obvious as those of other high-level languages, which made many students doubt whether the program was correct or not. As a machine-oriented programming language, assembly language was more efficient than other languages in calling resources. Therefore, the key problems such as course motivations, teaching methods, practice teaching and course assessment reform were discussed in this paper, and the reform ideas and implementation plans were put forward [1].

2. MOTIVATION

Nowadays, the more and more widely used high-level programming language led many students to think that learning assembly language was useless, so there was no enthusiasm and motivation for learning assembly language. In fact, assembly language was an efficient language that could give full play to the characteristics of computer hardware. It often appeared in some special occasions, such as data encryption, decryption, analysis and virus attack prevention. Moreover, learning assembly language can help students to better master the principles of computer work,

can improve students' ability to analyze and solve practical problems. Assembly language was a low-level machine-oriented language, which was the closest programming language to computer hardware and was high executive efficiency [2]. However, because assembly language was machine-oriented, it was necessary to have a deep understanding of the computer to program in assembly language. The main task of the course "assembly language programming" was to describe the structure of a computer with a computer model, and taught the lexical, grammatical and pseudo-instructions of assembly language, the representation of data in assembly language, the instruction system of assembly language, the basic control structure of assembly language program and its programming skills, the design method of subroutines, the design method of input-output and interrupt programs, Method and skill. This course enabled students to master the basic concepts, methods and skills of assembly language programming, and cultivate students' ability of reading, analyzing, designing and debugging assembly language. Assembly language had a large number of instructions, including nearly a hundred instructions, strict grammar rules, and needed to be familiar with hardware structure. Its learning process was abstract, and students generally reported that assembly language was difficult to learn, and it was difficult to generate interest in learning. How to solve these problems? After many years of teaching practice, we thought about the improvement measures, and carried on the implementation, also obtained the good effect.

3. TEACHING REFORM IDEAS

The core of teaching reform of assembly language programming was to change the current simple way of teaching, which stimulated students' interest in learning, cultivated the students' ability of independent thinking and practical ability, transformed a single explanation of knowledge into one that emphasized both the understanding of principled knowledge to understand and grasping the section between the knowledge points, the link between the discipline course, and theory with practice of thinking, so as to improve the students' practice ability and innovation ability. At the same time, some background knowledge was appropriately introduced in teaching, and practical cases

were used to put forward inspiring or open questions to encourage students to carry out active and exploratory learning. In the teaching process, in addition to emphasizing students' understanding and mastery of the principle knowledge and methods, it was more necessary to guide students to apply the mastered knowledge to real life and solve practical problems, so as to improve learning interest and enthusiasm.

4. PRACTICAL MEASURES

4.1. Length

4.1.1. Assembly Language is Closest Programming Language to Machine Language

If machine language was the essence of computer operation, assembly language was the closest one to it. The assembly language operation was directly oriented to the hardware, so when using the assembly language, students can perceive the operation process and principle of the computer, thus forming a clear understanding of the connection and interaction between the computer hardware and the application program. And these were the most able to exercise students' programming thinking logic, students who mastered these can form a soft, hard programming knowledge system. Compared with complex high-level language, assembly language instruction set was simpler, instruction operation was more direct, and learning from assembly language was more consistent with the principle of gradual learning. Therefore, for computer technology beginners, the importance of assembly language was irreplaceable.

4.1.2. Assembly Language is the Basis of Computer Technology

Assembly language can help students understand the high-level language better, especially the C language in the high-level languages. The operations on memory assembly language were based on memory addresses, while most troublesome concept for learning C language pointer was simply memory addresses. The most troublesome thing in the learning and application of pointer was to establish the thought mapping between the abstract concept of pointer and the actual memory unit, which was exactly what teachers often talk about in assembly language teaching. C language data types, parameters, function calls, global variables, local variables and other concepts and operations, teachers can use some of the assembly language operations associated with it. These abstract concepts and processes were formed into a specific image through assembly language to help students truly learn C language well. With the foundation of assembly language and C language, students can learn the following object-oriented programming language more smoothly.

4.1.3. Operation of Assembly Language is Related to the Hardware Closely

Many hardware installations use assembly language for embedded programming, because assembly language is more direct and efficient. Although many embedded development programs are written in C language, there are still some programs that C language cannot do. For example, not all registers in an ARM processor are addressable. When a certain register is not addressable, it cannot be operated by C language, such as CPSR in ARM. If embedded development programs do not use assembly language, many functions cannot be achieved [3].

4.2. Focus on the Demonstration in the Teaching Process

In the teaching process, teachers should insist on selecting some representative examples to demonstrate the program in class. Through the program code display and the program execution, the students can have a direct impression. For example, that students can use the DEBUG tool to understand the machine codes with different addressing methods, so they can understand what the immediate operand was and what the in-memory operand was. In the demonstration, the source program was compared with the results by disassembly, so that students can recognize the difference between the instruction and the pseudo-instruction, and there were no pseudo-instructions in the execution file.

4.3. Pay Attention to Students' Self-Learning Ability

This course involved a lot of knowledge, a lot of content to master, the terms and technical principle. It seems that some reluctant for the students understand in a class, so teachers should be focused on the interpretation of ideas in class, let students could learn themselves after class through various channels, and solve that were not yet understood, so as to cultivate the students' self-learning ability. Certainly, we should adopt prompt incentives to help students overcome their fear and tell them that in future work, students can only rely on themselves rather than teachers to overcome the difficulties of work.

4.4. Give Full Play to the Role of Practice

Debug tool is a debugging tool that comes with the Windows operating system. In it, according to the execution of each assembly instruction, you can see the execution result of each instruction, as well as the impact on each flag register and the change of memory unit. Therefore, students can understand the execution steps and functions under the instruction clearly. They can learn to use Debug tool to execute program segments, and check the changes of memory location by relying on the instructions provided by Debug [4]. For example, students can fully understand the difference between direct addressing and register indirect addressing in assembly language and the relationship between operands and addresses by Debug tool.

4.5. Improve the Course Assessment Method

In the past, the assessment of academic performance was based on only one test paper after the end of the subject. Therefore, according to the characteristics of assembly language programming, we adopted the following compound assessment method: The total score was the sum of the usual score, the experimental score and the final score, in which usual score accounted for 30%, experimental score accounted for 20%, and final score accounted for 50%. The usual performance mainly obtained from these several aspects including the student's attendance situation, the topic answer question situation and the topic quiz result. The traditional experimental examination only relied on the paper experimental report as the main basis of achievement, which caused students to plagiarism, similar work. This cannot reflect the phenomenon of students' practical ability. In order to avoid it, we reformed the traditional mode of experimental examination, the experimental performance was decided by experiment reports, computer experiment, and attendance together. Furthermore the description and the flow chart of the corresponding program algorithm must be given in experiment reports. An additional computer test was added. If the students can complete it, they can get more score. The final examination was also changed from the traditional written examination to the computer examination, giving full play to the role of information technology, and at the same time, it was more convenient to check the students' mastery of the course content.

5. CONCLUSION

Above was our point in the process of assembly language programming of teaching experience. Teaching is a huge project, which needs the efforts of teachers and the cooperation of students. To train students to master programming techniques, proficient in computer operation and program debugging techniques, there is still a long way in the future. We will continue to work hard!

REFERENCES

- [1] Zeng Yun, Wang Lixin, Si Bingbing, “*Consideration and Exploration on Teaching Reform of Assembly Language*”, *Computer Education*, 2015(13): 60-62.
- [2] Fuxiang G., “*Assembly Language Programming*”, Northeastern University Press. 2018: 2-9.
- [3] Brian R. Hall, Kevin J. Slonka, “*Assembly Programming and Computer Architecture for Software Engineers*”, China Machine Press. 2019: 80-99.
- [4] JohnSocha, “PeterNorton. *Assembly Language For The PC*”.Brady Publishers Inc. 1998: 102-123.