

A Study of Teaching International Students Java Programming at Shanghai Dianji University

Yuan Ren, Shunru Ji

School of Electronics and Information, Shanghai Dianji University, Shanghai 201203, China.

Abstract

"The Java Programming language" is a fundamental course for software engineering majoring freshmen at Shanghai Dianji University (SDJU). This paper describes the aims of this course, investigates students' characteristics, optimizes the contents of this course, and comes up with appropriate teaching strategies. Due to objective reasons, it is quite difficult to design and teach this course at a university effectively. A tentative plan is proposed make this course more useful and less boring so that international students are able to comprehend the theories and gain the programming skills better. Future work includes further evaluations of the proposed teaching method and applications of this method to other related courses.

Keywords

Java; Programming; International Student; Object Oriented; SDJU.

1. Introduction

"The Java Programming Language", also known as "Object Oriented Programming (OOP) in Java" is a compulsory courses for computer science majoring students. Since the international program of software engineering was launched in 2016, this course has also been offered to international students from overseas in either in the very first semester or in the second semester after the C programming language was taught.

This course is mostly similar across Chinese universities, although various colleges may focus on slightly different contents and may adopt their own teaching methods. However, by far there has not been any standard for this course offered to international students because of their distinct cultural and educational backgrounds, learning capabilities, and English proficiency levels. It is posing a serious question how this course should be organized and taught, and how this course can be optimized to be useful and helpful for foreigner. There have been a plenty of innovative discussions on teaching and constructing the course "The Java programming language" [1-2]. But few researches have focused on the pedagogical methodologies for international courses [3-4]. And few researches focus on teaching Java based on the prior knowledge of C [5-6].

This paper, based on our two year course construction program for international students at SDJU, endeavors to work out a suitable solution for teaching international students "Java Programming". The remainder of this paper is divided into five sections: section 2 describes the purpose of this course; section 3 analyzes the characteristics of international students; section 4 presents the course contents optimized on the basis of the previous analyses; section 5 discusses our corresponding teaching strategies; and section 6 concludes the paper.

2. Purpose of Course

The course "The Java Programming Language" teaches the basic Java syntax and introduces the essential concepts of OOP. The course comprises of 48 lectures and 16 labs of 90 minutes each. All international students majoring in software engineering will take this course in their first or second semester, depending on the program curriculum schedule. It aims to give students a fundamental understanding of OOP in the Java language, enable them to solve problems by writing programs, establish a foundation for future courses such as data structures and other programming languages, develop computational thinking, and prepare them for future careers. This course is one of the most

important courses for the software engineering program. It is expected that students can apply the knowledge and skills gained in this course to many subsequent courses.

3. Characteristics of Students

A majority of the international students at SDJU are from underdeveloped countries including Africa and Southeast Asia. Unlike domestic students admitted through national college entrance exams, international students have greatly different education backgrounds and proficiencies in computer. It is quite a challenging job to organize this course for a class of students of so various levels.

As far as learning habit is concerned, international students are totally different from domestic ones. They are much more active in and after class, like expressing themselves, asking questions, and sharing ideas with others. They are not shy or afraid to answer questions or discuss in class. Teachers will seldom feel bored or embarrassed after raising difficult questions. They also love to communicate with their teachers on social media apps. However, they do not always adhere to the regulations of the university or the rules set their teachers. For instance, punctuality and attendance are very hard for them to follow. Coming late, leaving early, and being absent are common among students of all majors. Instead of taking notes on notebooks or textbooks which is usually required by Chinese teachers, they prefer to record what they see and hear using phones. Chinese teachers always wish to think up efficient ways to give them lectures and help them pass exams.

4. Organization of Course Contents

Java is one of the most popular programming languages in the computer industry thanks to its large amount of functionalities. Java is an OOP language, but it can also be used for procedure oriented programming. Java has a huge collection of build-in APIs and therefore supports network, database, GUI, and multithreading programming as well as mobile app development. Java is such a big powerful language that it is hardly possible to cover every aspect of Java in a college course, so it is of great importance to choose the proper contents and to focus on the core part according to what courses the students have taken before and what course they will take next based on this course.

Previously, this course was scheduled in the very first semester of the freshman year. Newly admitted students learned Java as their first language with little prior knowledge about programming. This course should focus on basic programming theories and approaches including the use of variables, data types, selections, loops, arrays, and methods (functions) and algorithm design. Students were expected to understand how to solve questions by writing programs, master basic Java syntax and form computational thinking. Although Java is an OOP language, the concept of OOP and its difference from procedure oriented programming are not so important to new students with no programming experience. Java was mainly a prerequisite to many other courses. The students must master the fundamental Java grammar and algorithm designing skills in order to take other courses based on Java, such as the course data structures which was described totally in Java.

After several years, the program curriculum was innovated. In the new schedule, this course is moved to the second semester of the freshman year and will be offered after the C programming language. The origins of Java can be traced back mainly through the C/C++ languages, and Java therefore has a lot of similar components as in C, such as aforementioned variables, data types, flow control constructs, arrays, and function definitions and invocations. So after the students have taken the C language course, the Java language course unnecessarily has to repeat the similar part, or at least should not focus on that.

After a simple introductory session which helps the students prepare software and get to know this language, this course could quickly skip through the first several chapters and start discussions from where Java gets clearly different from C, for example, objects and classes. Meanwhile, Unlike C which is a procedure oriented, Java is purely object oriented. Besides the basic Java syntax, another emphasis should be on the concept and the approach of OOP. So this course focuses more on the four principles of OOP: inheritance, encapsulation abstraction and polymorphism. Because the courses of

networks, operating systems and database are offered in the second year or later, this course will not cover the advanced sessions for network, or multithreading database programming, and GUI programming is optional for self-learning. Exception handling, files and I/O are briefly introduced with several practical examples to follow and modify for improvements and extensions.

Cultivating the students' abilities to design algorithms and computational thinking is still an essential task for this course. Students are expected to transfer to Java quickly based on their acquired knowledge of C. Upon accomplishing this course, they will be able to master more programming languages smoothly. In the future, after they have taken those courses mentioned above, they can easily pick up where they had left off in this course and continue with the remainder aspects of the Java language. So in this course, we are never pursuing to cover everything with a limited number of lectures, instead, we focus on what the students need to know for taking the follow-up courses. This is a compromise between the "large body" of the Java language and decreasing number of class hours for this course.

5. Teaching Strategies

Based on the above discussions on the contents of the course and the characteristics of the students, the following teaching strategies have been adopted at SDJU:

- (1) Establishing students' interests in this course by introducing IT related to Java programming language, such as mobile app development.
- (2) Avoiding purely theoretical knowledge which does have applications in a short term.
- (3) Increasing the amount of practical and attracting examples, exercises and lab questions.
- (4) Extending the discussions on computer science to information technology and cell phones.
- (5) Providing a quick learning guide to help students transfer from C to Java fast.
- (6) Sparing students more time to discuss, express, and explore in and after class.
- (7) Providing students with knowledge summary notes and review sheets.
- (8) Assigning tasks involving innovation and creativity, such as writing challenging programming with knowledge or skills beyond requirement of the syllabus.
- (9) Conducting completely computer based final exams, and offering model paper before exams.

6. Conclusions

This paper analyzes the challenge to teaching international students "The Java Programming Language". The difficulties arises from both the course contents and its audience. The authors have been teaching this course at SDJU for five years. According the experiences gathered previously, the course is redesigned so as to be taught more smoothly based on the knowledge from prerequisite courses such as the C language syntax. Pure theories without much practical applications are largely replaced by attracting programming examples. This renovated course centers on students' abilities in solving problems by Java, an OOP language. Proper teaching strategies are adopted to accomplish the goals of this course, and to deal with students with various cultural and educational backgrounds. It is expected that this constructive plan will actually make this course a good transition between C and more programming related courses. Future work includes further evaluation of the construction scheme proposed in this paper according to the students' performances and feedbacks. The application of the cumulated experiences to other courses is also an area for future researches.

References

- [1] C. Yang: Research on Java programming course based on CDIO and iterative engineering teaching pattern, Recent Advances in Computer Science and Communications, Vol. 13(2020) No. 3, p. 519-530.
- [2] F. H. Cao: Reform of a Java curriculum based on a CDIO active project-driven method, World Transactions on Engineering and Technology Education, Vol. 14(2016) No. 1, p 129-135.

- [3] Y. Zhao: Individualized teaching and teaching process evaluation for international trade students based on information techniques, *International Journal of Emerging Technologies in Learning*, Vol. 14(2019) No. 23, p. 221-232.
- [4] M. Wu, E. Gide: Strategies to teach information technology courses to international students as future global professionals: An Australian case, *Procedia - Social and Behavioral Sciences*, Vol. 2(2010) Issue. 2, p. 4793-4799.
- [5] J. Su, W. Y. Weng, Z. B. Wang: A teaching path for Java object oriented programming, *Proceedings of International Forum on Information Technology and Applications*, Vol. 3(2009), p. 465-468.
- [6] I. S. Montagner, R. C. Ferrao, E. Marossi, F. J. Ayres: Teaching C programming in context: a joint effort between the computer systems, embedded computing and programming challenges courses, *Proceedings of Frontiers in Education Conference(2019)*.