

Analysis on the Teaching Reform of Programming Courses of the Specialized Computer Science in Agriculture and Forestry Universities

Zhang Ying

Department of Computer Science, College of Science
HuaZhong Agricultural University
Wuhan, China
zy@mail.hzau.edu.cn

Corresponding Author: Ye Yingze*

Modern Education Technology Center
HuaZhong Agricultural University
Wuhan, China
yyz@mail.hzau.edu.cn

Abstract—At present, there are many problems in the core programming curriculums of the specialized computer science such as low coordination with other courses, insufficient fusion and poor integration with the overall development of special discipline. The present study is mainly focused on programming curriculums for specialized computer science and the specific reform scheme is also put forward, with some innovation and practicability.

Keywords—*Programming; data structure; algorithm; practical teaching*

The undergraduate students majoring in the computer science in agriculture and forestry colleges and universities are required to complete "C / C ++ Programming Language" in the first academic year and "Data Structure", "Algorithm Analysis and Design", "JAVA Programming Language" , "Windows Programming (VC) ", which are programming courses and core foundation courses for the specialized computer science in the second academic year. It is more essential to cultivate students' thinking method in programming and as well as the ability of applying the programming language in solving practical problems than only to lay the foundation for the subsequent courses. In addition, the mastery of these courses will also affect the applications of computer skills in other fields[1, 2]. In light of this, the teaching quality of these courses attracts great attentions from both teachers and students.

However, there are some misunderstandings concerning the teaching of programming courses in most of the colleges and universities, which restricts the teaching of computer science. Therefore, it is extremely urgent to establish new reform ideas and to propose effective measures.

I. ANALYSIS OF CURRENT STATUS

In the rapidly developing information age of 21st century, it is necessary for undergraduate students to master some computer skills and computer professionals are also in urgent need in the competitive society. It is thus required that the teaching of computer courses in colleges and universities should be of quality and that it should be in line with the actual industrial technologies. However, after years of exploration, it is found that the teaching of computer courses still have some

problems in the agriculture and forestry colleges and universities.

The main problems are manifested as follows: there are many overlapping knowledge points among language courses; the relationship among courses is unclearly marked; course design does not consider the issue of consistency and transition; curriculum arrangement and teaching contents are not well connected with the special disciplines; course contents lag behind the actual requirements of the society. These problems restrict the cultivation and improvement of students' programming ability majoring in computer science.

II. REFORM MEASURES

After years of exploration and practices in the computer programming curriculum of Huazhong Agricultural University, a set of distinctive reform measures has been formed, with some achievement achieved in the preliminary implementation of these reform measures.

Due to a great diversity of computer programming languages and the complex relationship among these languages, the repetition of courses without consistence can be easily resulted if the languages are not well classified systematically according to social requirements. Concerning the computer science in our university, the programming language courses are relatively few with inadequate infusion and unclear curriculum structure. To address the above two extreme issues, the open and hierarchical curriculum system which is well classified is put forward.

A. Build the well classified sub-course group of programming languages

According to the professional direction of students and social requirements, the classified curriculum system is built to achieve the vertical integration among different programming languages. Currently, the mainstream development tools in need can be divided into two types: C ++ or JAVA, which are required for students majoring in the computer science. As a result, they have not enough time to do more practices and learn both courses well, and it would be a waste of energy. NET and J2EE are most frequently used in the actual application systems. However, students are required to study

most of these advanced programming languages by themselves. The problems concerning the current curriculums include and the logical among different courses is not clarified with many overlapping knowledge points. In addition, students do not usually have the opportunities to get in contact and truly understand some of the mainstream programming languages such as COBOL, Python and Perl at school. Therefore, the students lack necessary knowledge of these languages, with reduced competitiveness in employment.

Therefore, the language courses are divided into three categories in the classified curriculum system. Category I is composed of C / C ++, VC, C # and ASP.NET; Category II is composed of C / C ++, JAVA, JSP and J2EE; Category III is composed of other programming language courses. Among them, the courses in Category I and Category II are alternative compulsory courses for students while the courses in Category III are elective courses.

The purpose of including C # and ASP.NET or JSP and J2EE in Category I and Category II of advanced courses is to cultivate computer professionals with programming ability and good mastery of a specific programming language by setting up the advanced programming language courses according to the requirements of society. This is also based on the fact that most of the graduates newly recruited by companies have to subject to pre-post training. These courses aim to strengthen the practical ability and problem-solving ability of students through case teaching and engineering project teaching.

It is found in the survey that programming languages used by large companies such as PHP, COBOL, Python, Perl can be learnt by students through self-teaching after the their mastery of basic C / C ++ programming skills; or the elective practical courses can be included in teaching programs in universities. The cultivation of programming language abilities in this category not only can expand students' scope of knowledge and improve teacher professionalism and ability, but can also include the top 5 to top 20 programming languages in the world into the teaching program in the form of elective practical courses, which is quite an innovative measure. These elective courses are arranged in the fourth or fifth semester, and they are cut into specific practical modules in light of the large-scale application projects. The programming language can be mastered through practices on the computer as designed in the module. The reform of teaching contents and teaching model not only broadens the horizons of students but also improves students' practical ability.

B. Build the hierarchical sub-course group of programming languages

The knowledge points of each course are decomposed and then integrated according to the relationship among knowledge points so that the hierarchical and comprehensive objectives of ability cultivation can be reached.

With respect to the building of hierarchical curriculum system, it is required that the knowledge points in different courses to be decomposed and then to integrated in order to make clear the relations among curriculums. In this way, the students can understand the associations among knowledge points as well as the progressive relations of the knowledge

points that have been learnt. This system helps to lay foundation for the subsequent course planning and the integration of course contents.

The main knowledge points of basic language courses such as C / C ++ and JAVA are as follows: data types, functions, arrays, pointers, structure types, files, classes, events, network programming, and so forth. For example, data types, functions, arrays and other knowledge points are included in both C and C ++ and files, classes and other knowledge points are included in both C ++ and JAVA. Although the grammar rules and description methods as well as emphasis vary from language to language, the essential contents of knowledge points are similar with many overlaps. It is necessary for teachers to choose contents of teaching rather than give scripted lessons. Furthermore, in order to ensure a good coherence and transition of knowledge points among different courses with well organized and non-repeated contents, a teacher can be fixed to give lessons from the first phase to the third phase.

Secondly, the study of programming ideas should follows such a line, i.e. various programming development models based on different applications as it transits from process-oriented, and object-oriented and further to application framework-oriented are the reflection of the hierarchical course system aimed at the cultivation of programming ability. First, the basic programming ability is the foundation. Only with the mastery of writing program code can students be able to solve problems using the programming method. The study of C language which is oriented towards process is enough to cultivate such qualities. Second, the application ability is reflected by the students' ability to solve practical problems by applying the knowledge learnt. With the mastery of such comprehensive design capability, students can "apply their knowledge." After the study of C ++ language which is oriented towards objects, the students can choose either JAVA or VC, and C # and ASP.NET or JSP and J2EE which are more engineering, to achieve a rapid improvement of skills. Third, through the study of other frameworks or various programming language models based on different applications, the students' development capacity will also increase day by day to meet the requirements of the society. At the same time, innovative thinking and self-learning ability of students is essential for further study and research, and is also the essential training objective.

C. Build the open teaching practice platform

The teaching practice platform is built based on the open of time, place, content and interest. The practical projects are scientifically and reasonably set up according for different courses. The students select subjects via ISAS system developed by our department after forming team at their own will at the beginning of lessons. The selected project is the subject throughout the entire course. Each team is required submit weekly progress reports and attend project reply at the end of the course. Before each experiment, students are required to complete the preview test via on-line preview subsystem (question bank for preview test is primarily designed to test verification knowledge). During the experimental teaching process, the students design and complete the related practical projects independently by consulting the background

knowledge with the use of experimental auxiliary subsystem under the guidance of the teachers. After the experiment, the teacher will comprehensively evaluate the experiment performance of students according to the preview test scores, sincerity, single test scores, scores added for elective courses, project reply and experiment assessment to achieve the process-oriented diversified assessment.

By establishing the open teaching practical system, the original teaching mode is no longer in use. The new teaching model featured by the open of time, lab, content and interest transforms students into active explorers of knowledge from passive recipients and teachers into the culturists of interest, abilities and quality from mere imparters of knowledge. By this means, the emphasis is shifted from on generality education to individuality education and the positive interactions among teachers, students and teaching practice platform are created.

D. Build sub-course group of algorithm courses

The various algorithms commonly used in different special disciplines are extracted to form an algorithm library. Then the algorithms in the algorithm library are connected with the specific core curriculums so that the specific algorithms in the algorithm library can better serve their application in special disciplines.

The main purpose of computer programming is to solve the actual application problems. First, a logical model is abstracted based on problem analysis; next, the basic operations in this model are given; then the optimal data structure is determined to represent the model; finally, the programming simulation is prepared and actual problems are solved. During the programming process, algorithm design is the key part as it highly relies on the data structure while the selection of data structure depends on the requirements of the problem itself. Algorithms are closely connected with data structures. Wirth, the father of Pascal language, wrote a book titled "Algorithms + Data Structures = Programs", indicating that the importance of algorithm and data structure to the program design is self-evident.

The data structure courses mainly include four logical relations, namely, table, tree, graph and file structure; four types of storage methods, namely, sequence, chain, index and hash; and two main operation searches and sequencing. Algorithm analysis and design program curriculum mainly include: the classic algorithm design strategies such as divide and conquer algorithm, greedy algorithm and dynamic programming; advanced algorithm design strategies such as randomized algorithms and approximation algorithms. The courses of Data Structure and Algorithm Analysis & Design for students majoring in computer science in many colleges and universities are two professional required courses. Algorithm Analysis & Design is included in the curriculum for postgraduate students in some colleges and universities. In other colleges, these two courses are directly united into one course, i.e. Data Structures & Algorithm Analysis. Regardless of the arrangement, it is agreed by our curriculum group that the organic integration of Data Structure and Algorithms Analysis as well as introduction of algorithms in teaching data structure can better mobilize the enthusiasm of students with

better teaching effect if the practical application cases are also taught.

For example, Prim's algorithm and Kruskal's algorithm of minimum spanning tree in the graph theory of data structure is a good case applying greedy algorithm in the analysis courses. Here we analyze the algorithm in detail through instances of data structure so that the students can understand and master it in advance. For another example, the game of exchanging seals among children is taken as a case to compare and introduce various algorithms. It is assumed that the weights of the seals are 1g, 5g and 11g, respectively, which shall be exchanged with seals with the weight of 15g in the minimum number possible. As long as the greedy algorithm is concerned, 1 seal of 11g and 4 seals of 1g, i.e. a total of five seals are required. With dynamic programming method, only 3 seals of 5g are required. It can be seen that though the design idea of greedy algorithm are easy to understand and questions are easier to be solved. However, the optimal solution can not be necessarily obtained for some problems using the greedy algorithm.

First of all, the cases selected should come from everyday life, to arouse the learning enthusiasm of students; secondly, the students are guided to consider the question of how the results may vary with different algorithms. Thus their interest in learning interest is increased and their awareness and ability to identify, analyze and solve the problems are cultivated; thirdly, due to reform and adjustment of teaching methods and contents, the inflexibility and monotony of the old teaching method and content will disappear. Thanks to the integration of algorithm design and data structures, the teaching effect will inevitably reach a higher level.

E. Integrated algorithm, professional courses and special discipline

In order to show that the algorithm comes from the professional courses and serves the special disciplines, the relevant algorithms required for the professional courses are abstracted by our curriculum group and they are taught in advance through case teaching to provide foundation for subsequent study of professional courses. In addition, the algorithms are classified, forming a colossal algorithm library according to the four special disciplines in our department: graphics & images, bioinformatics, internet of things and software engineering. According to the integration points of special disciplines and specific algorithms, the specific project modules are abstracted; specific cases are presented in combination with the large integrated practical training platform.

If one algorithm in the algorithms library is originated from one or more core professional courses, it also serves the specific applications in one or more special disciplines. The algorithm-based systematic research and teaching will facilitate the improvement of students' professional capacity and competence.

For example, apart from being involved in the optimization of image coding in professional course "Computer Graphics", the KMP algorithm is also connected with certain applications in some special disciplines such as graphics and bioinformatics. Take the data mining algorithm as another example. It is not

only involved in the computer science courses "Operating System" and "Computer Network" , but also associated with some specific applications in the special discipline "Internet of Things". Through analysis and comparisons supplemented by explanations of real cases, students will have a more intuitive and profound understanding of the corresponding.

III. SIGNIFICANCE OF REFORM

The building and optimization of sub-course groups of programming language courses, scientific positioning of these courses and correction of objectives will contribute to the elimination of barriers between courses as they are based on the holistic approach. The reform will enhance the association among courses and facilitate the integration of theory & practice, knowledge & ability.

The systematic research on data structure and sub-course group of algorithm courses is focused on the integration of algorithms with professional courses and special disciplines respectively through the integration of knowledge points of courses on data structure and algorithm analysis. This integration will be helpful to improve the students' ability of algorithm design and to have a more clear understanding of the algorithms in the professional core courses. Also, the practical ability of the students in line with professional development goals will be fostered so as to achieve the objectives of cultivating practical computer professionals.

The study of open experimental teaching platform gives the students opportunities to have contact with the actual problems in engineering so that the students' innovative awareness and

ability can be enhanced. At the same time, it is significant to promote training in scientific research and academic competitions for undergraduates and to fully mobilize the enthusiasm of colleges, students and communities to participate. Moreover, it is highly beneficial for building research and exploratory universities, increasing the number of students benefiting from it as well as enhancing students' practical and innovative ability.

ACKNOWLEDGMENT

This work was financially supported by the Fundamental Research Funds for the Central Universities(Program No. 2011JC018).

REFERENCES

- [1] Computer Science and Technology Education Advisory Committee of Colleges and Universities of the Ministry of Education. Teaching Implementation Program of Core Curriculum for Computer Science and Technology Specialty in Colleges and Universities [M]. Beijing: Higher Education Press, 2009
- [2] Computer Science and Technology Education Advisory Committee of Colleges and Universities of the Ministry of Education. Development Strategy Research Report and Professional Norms for Computer Science and Technology Specialty in Colleges and Universities (Trial) [M]. Beijing: Higher Education Press, 2006
- [3] Li Weiqiang. Analysis of computer basic course teaching in colleges and universities [J]. Three R's (Education and Teaching Research), 2011,5:164,52
- [4] Zhang Yahong. Discussion of teaching method of computer programming curricula [J].China Science and Technology Information, 2007, 19:250-251.