

# Deeper Learning Cycle (DELIC) Practiced in Computer Programming

Chaoyan Zhu\*

Computer science

Ningbo Institution of Technology, Zhejiang University

Ningbo, China

zcy@nit.net.cn

\*Corresponding author

Genlang Chen

Computer science

Ningbo Institution of Technology, Zhejiang University

Ningbo, China

cgl@nit.net.cn

**Abstract**—Computer programming courses aim to develop students' programming skills and problem solving abilities. But it cannot easily be achieved through traditional teaching method which focuses on boring grammar more than approach and thinking. Students should learn deeper. In this article, we practiced Deep Learning Cycle (DELIC) in the curriculum of C programming language. In the 7 steps of DELIC, we focused on step 6, processing the learning deeper. We adopted task-based mode and online judge like ACM-ICPC, but designed our own problem repository with teaching contents closely combined to make students learn deeper. Practice has proved that, based on DELIC teaching model, students' programming skills and proficiency have been greatly improved.

**Keywords**-component; Deep Learning Cycle(DELIC); ACM-ICPC; Online judge; Task-based

## I. INTRODUCTION

Computer programming courses are very important and mostly compulsory to CS students. These courses not only require students to master related theory, but also develop students' programming skills and problem-solving abilities. But it cannot be easily achieved. Students can't understand concepts clearly, and it's hard for them to grasp the internal relations between concepts and to apply to the application after learning[1]. Eventually many students lose their interests in the course or even in the major.

Deeper learning Cycle(DELIC)is proposed by American famous teaching reform expert Eric Jensen and Leann Nickelsen. Deeper learning is different from simple learning which characterized by mechanically and passively acceptance of knowledge. Deep learning lays emphasis on active learning and critical learning. It requires students to establish the internal connections between new learning knowledge and existing knowledge on the basis of fully understanding the meaning of study contents, learn to migrate existing knowledge to new situations, and ultimately to make the right decisions and solve problems.

Deeper learning mode applied in programming courses can improve all students' learning literacy, develop students' active thinking way to explore and acquire knowledge, let students actively participate, willing to study, dare to experiment, diligent in thinking.

## II. PEDAGOGICAL FOUNDATION

### A. Simple Learning and Deep Learning

Simple learning is one-time learning, which requires no feedback or error correction. It doesn't take too much effort except for rote memorization. Memorizing important historical dates, the alphabet, learning a list of vocabulary words and a specific definition are examples of simple learning. It is obvious that simple learning serves as the basis for all future learning. But it lacks the complexity of thought and learning may be laborious.

Deeper learning requires us to follow multiple rules of operation to complete task. In other words, there are multiple thinking steps to achieve deeper learning. For example, when we're reading, doing multidisciplinary thinking, solving problems by devising solutions and creating goals and strategies to achieve those goals, we need deeper learning. Typically, greater understanding, greater retention, and more application of concepts and skills are possible when learning is deep.

### B. Deeper Learning Cycle

Most complex thinking is not hardwired, it must be taught. The Deeper learning cycle (DELIC) represents how each student can be challenged to arrive at the deeper level of learning. There are 7 steps in DELIC applying in daily lesson illustrated by figure1.

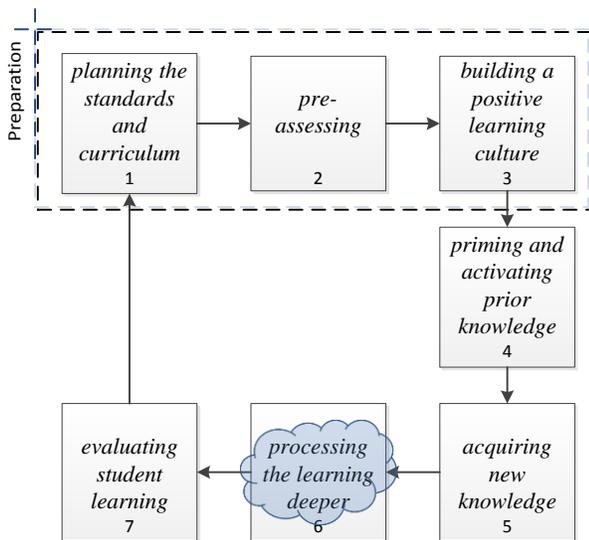


FIGURE I. SEVEN STEPS OF DELC

The first step is planning the standards and curriculum, which aims to design how to take the lessons and units for teachers[2]. The second step is preassessing to see what students know about the standards and objectives. The third step is to build a positive learning culture for student to help them learning successfully. The following three steps are priming and activating prior knowledge, acquiring new knowledge and processing the learning deeper. Among them, processing the learning deeper is the most important step for deeper learning and it's going to take far more time than in classroom. The last step is evaluating student learning.

### C. ACM-ICPC

ACM-ICPC(ACM International Collegiate Programming Contest), is an annual contest sponsored by ACM(Association for Computing Machinery), and aims to show students' innovation ability, team spirit and under pressure programming, analysis and problem solving ability[6].

During the competition, 7 to 10 problems need to be solved using C, C++ or Java by each team in 5 hours with 1 computer. Programs of each team will be submitted to the online judge servers, and will be judged as right or wrong and inform the team. The one who solved problems correctly the most and cost time the least will be the winner team.

## III. COURSE DESIGN AND DEVELOPMENT

There are many programming language curriculum, here we take C programming language as an example. On the one hand, C programming language is similar to other language courses such as English or Japanese which needs simple learning when remembering words, grammars. For example, there're keywords, expressions, statements and frequently used library functions in C language, and they need to be remembered. On the other hand, we need to solve problems by devising programs using C language, so we need divergent

thinking, creative thinking, and computational thinking, which can be gained through deeper learning. In conclusion, during the study of C programming, we need both simple learning and deeper learning, the first served to the latter, and the latter is critical in C language programming learning.

### A. The Previous Teaching Method

The traditional teaching method of C programming language is mainly classroom teaching, supplemented by computer experiments for verifying. Teaching contents will be organized according to the teaching material in depths and range. In the classroom, the knowledge of the language, such as grammar, will be explained by teacher through the blackboard or electronic slides. During the classroom teaching, teachers are apt to focus on boring grammar other than more import things such as program design approach and thinking, so it will ultimately lead to the phenomena of grammar learning and programming disjointed.

Furthermore, many students fell behind one after another during the course because of the lack of usual autonomous learning and practice. Finally, those lag behind struggled to pass the exam through temporary assault, and their exam result manifested that they maybe have mastered some details of C language, but failed to achieve the teaching goals, to enhance students' programming ability and problem solving ability.

### B. DELC Practiced in the Course

From the above, we have known that C language programming needs deeper learning more than simple learning. But most complex thinking must be taught. Many students lack the exposure, training, and guided experience to think at complex levels. So the introducing of DELC, will empower students to think deeply.

The first three steps in DELC are the preparing steps. Step 1 begins with the course standards and curriculum. In daily lesson, the teacher explains how to take the lesson and unit and create questions that explain the big picture of the objectives. Teacher must always have a guide for what they want students to accomplish. Here the teacher can design some problems as tasks related to the lesson for students to accomplish. The second step preassessing associated with step 1 is from the perspective of students to see what they know about the standards and objectives, so that teachers can know students' background knowledge better and decide where to start.

The third step is to build a positive learning atmosphere. Motivating students in a positive, motivated state of mind so they care about learning helps them dive deeper into their culture. Our e-teaching environment can serve this goal well, as displayed in figure 2. As we all know, deep learning is enhanced when students are highly challenged. So, this environment provides students the programming training system(composed of problem repository and online judge server) which is full of challenge and can greatly arouse their programming interest and encourage them to learn deeply, we will introduce it in detail in section 3.3. In addition, the environment can be regarded as a social learning network, students can take activities such as communication with

students and teachers, trouble shooting, academic discussion, resource sharing, completion of knowledge tests, etc. By these ways they can have a trusting relationship with teachers and other students.

Step 4 and step 5, priming and activating prior knowledge and acquiring new knowledge are always been of concern by teachers. The two steps usually happen in classroom teacher-directed.

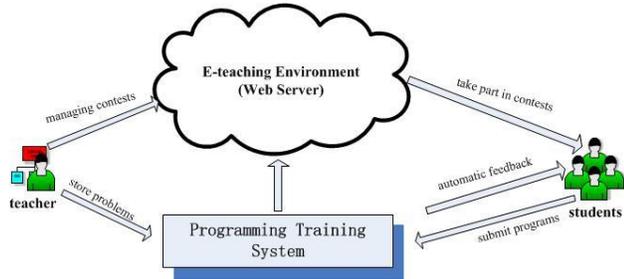


FIGURE II. STRUCTURE OF THE E-TECHING ENVIRONMENT

In the all 7 steps of DELC, the most important step is step 6, processing the learning deeper. Here, we adopt task-based mode like ACM-ICPC which changes teaching methods from knowledge-based traditional methods to task-based multi-dimensional interactive teaching methods. Task-based mode can help students process the learning deeper, however, how to design suitable tasks is the most important thing which should be implemented in the step 1 in DELC.

ACM-ICPC can provide guidance for us, we adopt online judge server like ACM-ICPC to judge whether the program submitted can pass or not. But contestants attending ACM-ICPC are always excellent students selected from the students, and the problems used in competition are too difficult for normal students, they greatly exceed the teaching requirement of the course of C programming. So the competition doesn't suitable for normal students, and can't promote the common students' programming ability effectively and thoroughly. So we can use the form of ACM-ICPC, but we should design problems and tasks suitable to common students who are newcomers to programming language.

As figure 3 illustrated, the teacher first design the problem repository which exhibit to students as tasks. Each task is corresponding to some knowledge points, students learn to master the knowledge background by programming and submitting programs to the online judge server. Using this mode, on the one hand, students' independently studying enhanced, on the other hand, students can gain satisfaction, sense of accomplishment, stimulate their interest in learning and arouse their main initiative greatly.

We also shouldn't forget evaluate students to know how they're going on, so we can have a better design of next lesson or unit. We accomplish the step mainly through the performance of students in training system and the results of knowledge tests.

### C. The Programming training system

The programming training system consists of a problem repository and a judge server illustrated in Figure 3.

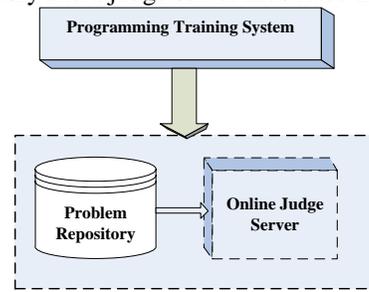


FIGURE III. PROGRAMMING TRAINING SYSTEM COMPOSITION

The online judge server is just like the function provided by the well-known OJ(Online Judge) systems such as SGU(Saratov State University Online Contester)[3], UVa Online Judge[4], OJ system of Peking University[5], Zhejiang University, etc. These OJ systems analysis are all based on B/S structure. Students use a web browser to submit source code, and the server receives the source code, compiles it, compares the run results with test data what is prepared together with the problems, returns the corresponding result of compiling and execution to the students. To stimulate the students' interest in learning programming, the online submission systems also integrates competitive encouraging mechanisms such as performance statistics, and ranking function, which not only enhance the interactivity but add a lot of fun.

Although these automated assessment systems have become popular for training programming skills, they are more suitable for competitors not for beginners. So problem repository closely combining teaching contents and appropriately designed plays an important role.

When we design the problem repository, we first give out the overall teaching goals, and then the overall goals are subdivided into small learning modules which are refined into tasks and showed to students.

For each task, a detailed task description, format requirements of input and output together with samples, and comprehensive test data are provided. The training system requires students to pass the test of all test data. If the main program is correct, but because of not taking the special circumstances into account and leading to some test data went wrong, then online judge server will judge the program as wrong.

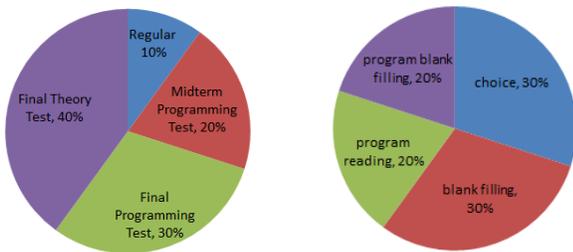
In addition, what knowledge and information are needed to complete the task is given out by teachers too. Students who haven't mastered the knowledge involved will fail to accomplish. In order to win, students will learn the related contents by themselves. The knowledge gained in this way is more profound than teachers teach in classroom, and they learn to learn deeply driven by the tasks.

We also consider the difference among students, so different objectives are for different students. There are tasks corresponding to basic objectives and higher objectives in our problem repository. The former tasks are required to complete for all students, the latter tasks are prepared for those who can finish the basic tasks easily. If students complete all tasks without any difficulty, we recommend them to directly participate in the famous OJs, which focus on algorithm training more, that is said open objectives.

#### IV. EVALUATION

##### A. Evaluation Mode

The core teaching goal of C language programming curriculum is to develop students' programming skills and problem-solving ability, therefore assessment methods are focusing on students' programming ability. The overall score of the course is composed by the following ratio: 10% of regular grade, 20% of midterm programming grade, 30% of final programming grade, and 40% of final theory test. Midterm and final programming exams are using the training system, students are required to complete 11 tasks within two hours; theory exams are composed of four kinds of question types: choice, blank filling, program reading and program blank filling, and the last two kinds accounted for 40% .



(a) Overall score composition (b) Final theory test composition

FIGURE IV. THE OVERALL SCORE OF THE CURRICULUM

##### B. Results

There are four classes majoring in CS in 2012 to 2013 school year. The average score of each class in midterm and final programming test are listed in table 1. There are 11 questions in the test on a 110-point scale. From the table 1, we can see that the class CS111 and CS112 do better than the class CS113 and CS114, and the class CS113 showed a great improvement in final testing than in midterm testing. The whole situation is: The course was taught by four teachers, one teacher took charge of one class. Only class CS111 and CS112 used the training system from the beginning of the semester to the end, CS113 used the system from the midterm exam to the end of the semester, and CS114 only used the system in the

final exam. From Table 1, it can be clearly seen that after using the system for training, student programming skills, problem solving ability have been greatly improved, and students considered the problem more fully, programmed more intricately.

TABLE I. AVERAGE GRADE OF PROGRAMMING TESTS IN 2012-2013

Results Class No.	Midterm grade	Final grade
CS111	81.5	93.75
CS112	85.58	96.75
CS113	46.67	95
CS114	Not Attend	48.11

#### V. CONCLUSION

After DELC practiced in the curriculum C programming language, the interests of students in learning programming are stimulated, learning enthusiasm and initiative are greatly improved, and students' programming skills and proficiency have been greatly enhanced. Students understand problems more comprehensively, and program more roundly. Practice has proved that, based on DELC teaching model, students learn deeper. This model can apply not only to computer science, but other engineering majors where C programming language is taught.

#### ACKNOWLEDGMENTS

The work described in this paper was fully supported by the Higher Education Reform Project in Zhejiang Province kg2013520.

#### REFERENCES

- [1] Jenkins, T., On the difficulty of learning to program, Proceedings of the 3rd Annual Conference of the LTSN Centre for Information, 4 (2002): 53-58, 2002.
- [2] Eric Jensen, LeAnn Nickelsen, Deeper Learning: 7 Powerful Strategies for In-Depth and Longer-Lasting Learning, Corwin press, 2008.
- [3] Saratov State University Online Contester. <http://acm.sgu.ru/>
- [4] MA Revilla, S Manzoor, R Liu, Competitive learning in informatics: The UVa online judge experience, Olympiads in Informatics, Vol. 2, 131-148, 2008.
- [5] Li Wen-xin, Guo Wei, Peking University Online Judge and Its Applications, Journal of Changchun Post and Telecommunication Institute, S2, 2005.
- [6] ACM International Collegiate Programming Contest. <http://en.wikipedia.org/wiki/ACM-ICPC>