

Types of Learning Objects

Ion Mierlus – Mazilu

Department of Mathematics and Computer Science
Technical University of Civil Engineering, Bucharest
B-ul. Lacul Tei, Nr. 122-124, Sector 2, Bucharest
38 RO 020396 Romania
mimi@mail.utcb.ro

Abstract

The Codewitz Learning Objects are interactive visualizations of program code examples or programming tasks. They have been developed to help students to understand programming structures more easily. A Codewitz Learning Object can cover any specific programming problem in any programming language. Learning Objects can also cover the problem-solving logic at the algorithmic level.

A learning object focuses on one specific learning goal. Each learning object has to be independent, without links to other objects or resources. Thus for example, server-side generated web pages are not valid as Codewitz Learning Objects. This independence ensures the real reusability of the learning object.

Keywords: *eLearning, Learning Objects, Distance Education*

1. Introduction

Learning computer science and especially programming seems to be a difficult task for students today. What is the best way to teach computer science to novice students is a question; many teachers have been considering recently and is even more relevant now, than a few years ago, when computer science no longer seems to be an attractive subject to university students in

Western Europe, America, Australia, New Zealand and even other parts of the world. What can we do to help students to gain better understanding of fundamentals of programming and feel the joy of running programming codes successfully? The answer is not simple and depends on what you consider most important in teaching. Some authors emphasize choice of first language, other emphasizes teaching methods and organization of the study and some mention the use of different media or multimedia for support. An object orientated approach or a procedural approach has been discussed and communication skills and collaborate skills are among many desirable skills computer science students should be trained in [Ma 2005].

Computer science students often have diverse backgrounds and learning styles that can call for miscellaneous learning and teaching methods. Some even state that “methods used to teach introductory computer programming to college students are becoming outdated” [McKeown 2004]. Research indicates that novice programmers have difficulty in understanding programming concepts, the syntax of codes and the interpretation of blocks of codes [McGill 1997]. Teachers are looking for new methods and support for their teaching, as they want to help and motivate their students, and learning objects with their visualization might be considered feasible support.

2. Distance education and learning objects

There are many definitions regarding the term of distance education, involving the educational access, closely related to the information technology and communication infrastructure.

The California Distance Learning Project's definition is: "Distance Learning (DL) is an instructional delivery system which connects learners with educational resources. DL provides educational access to learners not enrolled in educational institutions and can augment the learning opportunities of current students. The implementation of DL is a process which uses available resources and will evolve to incorporate emerging technologies."

As defined by Michael Moore, the then director of The American Center for the Study of Distance Education, Penn State: "Distance education is planned learning that normally occurs in a different place from teaching and as a result requires special techniques of course design, special instructional techniques, special methods of communication by electronic and other technology, as well as special organizational and administrative arrangements".

Distance education became significant because of its divergence from the common centralized school model by bringing the school to the student instead of sending the student to the school. Distance education became successful because it filled a need generated by an increasing number of nontraditional students. The potential audience for distance education is much more varied and much larger than any educational establishment estimated.

Today, higher education is a necessity for those who wish to work and prosper in an economy based on information manipulation, which is becoming dependant not on sheer muscle power, but on brainpower. Today, the workforce is rewarded for how well and how fast problems are detected and solved.

As Benset (2005) states the term learning object has been over-used and there are different definitions, strategies and standards of learning objects [Sun 2005]. IEEE has a broad definition of learning objects as "any entity, digital or no digital, which can be used, or reused referenced during technology supported learning." Nugent et al. (2005) declare a learning object simply as "a structured, standalone media resource that encapsulates high quality information to facilitate learning and pedagogy" [Nugent 2005].

The *Codewitz Learning Objects* are interactive visualizations of program code examples or programming tasks. They have been developed to help students to understand programming structures more easily. A Codewitz Learning Object can cover any specific programming problem in any programming language. Learning Objects can also cover the problem-solving logic at the algorithmic level.

A learning object focuses on one specific learning goal. Each learning object has to be independent, without links to other objects or resources. Thus for example, server-side generated web pages are not valid as Codewitz Learning Objects. This independence ensures the real reusability of the learning object.

Technically, the above definition restricts the choices for implementation. At the time of writing, these three suitable techniques are available: Flash, Shockwave and Applets. They allow Codewitz Learning Objects to be developed according to the above definition. Any solution having web-browser capability and the independence from any other technical aid is acceptable. Only browser aids like plug-ins or players are accepted.

Characterization of Codewitz Learning Objects

- Browser capable
- Stand-alone (no server or other technical dependencies)

- Reusability
- Not linked to any other learning object or resource
- Focusing on one specific learning goal

3. Codewitz learning objects

The students of basic programming courses usually do not make much progress. To improve the students' progress we start to produce and evaluate unique illustration, animation and visualization aids for students and teachers of computer programming, who are involved in the field of professional and/or higher education. By reducing complexity in learning computer programming these aids, referred as Learning Objects help the learners to better understand and master, and the teachers to better explain and illustrate the problems connected to the use of basic and advanced structures in computer programming.

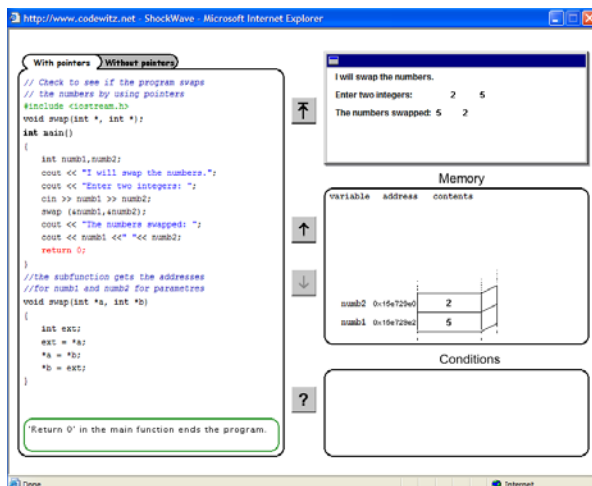


Figure 1

The learning objects discussed in this paper are from the Codewitz (www.codewitz.net) project which is a Minerva Socrates project that emphasizes developing and producing interactive web-based learning objects for programming courses.

The idea of the program visualization learning objects is debugger like step-by-step program execution in both forward and

backward directions (Figure 2). The program code is highlighted in each important step of the program execution and the run of the execution in code is also visualized by arrows when necessary. In each step of the program execution console is visible as well as the memory area. There are also areas for the conditions and for the short explanations of the current step. The memory part is the only one where the layout can be changed according to the subject as learning goal. These changes appear for example in case of arrays when the structure of the array is visualized.

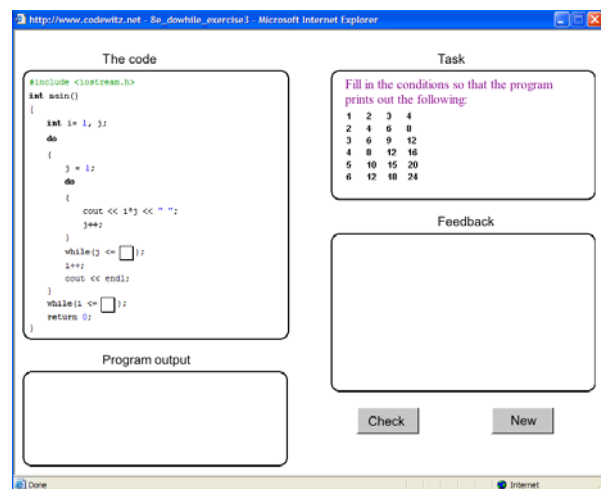


Figure 2

The design of most of the objects is similar to the one in Figure 1 but some have a slightly different design as can be seen in Figure 3, where the task is explained for an exercise and a feedback given.

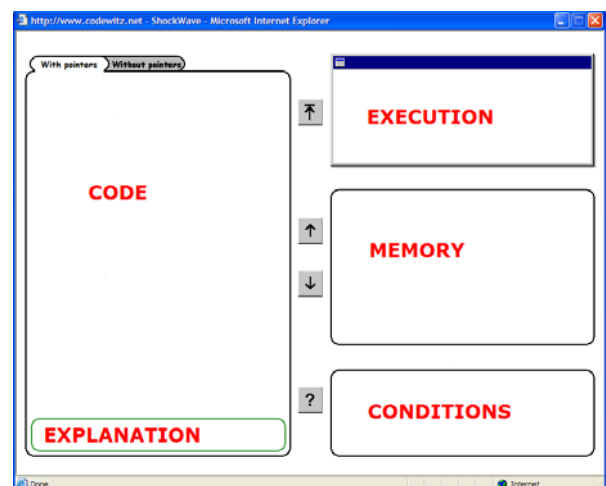


Figure 3

Learning objects in the Codewitz project are web-based standalone visualizations of programming tasks or code examples built for clear specific learning goals. The Codewitz learning objects are so far mainly for supporting C++ teaching and learning but some of the objects are also for teaching/learning Java. At the end of the project as many as 178 learning objects have been made and they are accessible through the project's website where pans for about 400 new objects can also be found. Figure 1 shows an example of a learning object which explains pointers. Here we can see that the object has an area for input/output from the student, execution that shows step by step what is going on and an area for Memory and Conditions. Many of the objects also have an explanation area.

To make the objects the partners could use different methods or programs and most of them used Macromedia Director so many of the objects need Macromedia Shockwave to run but some are made with Flash and some with Java.

44. A case study

To improve the progress of the students of programming language courses, visualization learning objects are produced and used as learning materials. In some institutions the students have different study backgrounds. Some come to the course with non-theoretical and some with theoretical studies behind them. The focus is set on the differences in the effects of program visualization learning objects on the students' course results.

The study was organized on the same course in two years: In the first year students do not have the program visualization learning objects as learning material available and in the second year they have the program visualization learning objects available. The students study exactly the same course.

The effects of the program visualization learning objects on the results are then analyzed by the final course points and grades and activity of the students and also with a survey about all learning materials available held at the end of the course.

The study was conducted in Technical University of Civil Engineering during the two years (2004-2006) for Civil Engineering students department who are non-major students in programming. These civil engineering students have one obligatory programming course included in their studies. This course is called Programming Language and the course covers the first steps of programming like variables, selection, loops, arrays and functions. The Programming Language course is timed in the second semester of the second year of their studies and the scale of the course is three credits. The course consists of lectures and lab exercises. Each student has two lecture hours and also two lab hours per week.

Each year four groups of around 25 civil engineering students are formed. The groups have the Programming Language course parallel with the exactly same content. It is assumed in the course everyone has no previous knowledge about programming.

The program visualization learning objects were occasionally also used as program examples by the teacher. The both year courses were held by the same teacher with the same material and with the same outside classroom assessments for the students. At the end exactly the same paper exam was given to the students in both years. The exam papers were not given back to the students. Thus the questions in the exam are considered not to be known by the next year students.

The results consist of the final grade and the activity of the students. Final grade is between 1 and 10 where 4 is failed, 5 first grade for the completion of the course and 10 is the best grade (Figure 4).

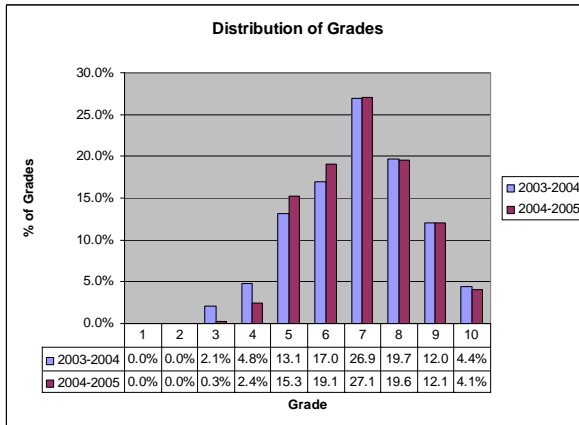


Figure 4

In the survey the students were asked about their programming skills before the course, about the skills in using computers in general, about all learning materials used in the course and program visualization learning objects (used only in 2004-2005 courses). The survey result about visualization learning objects was very good (Figure 5).

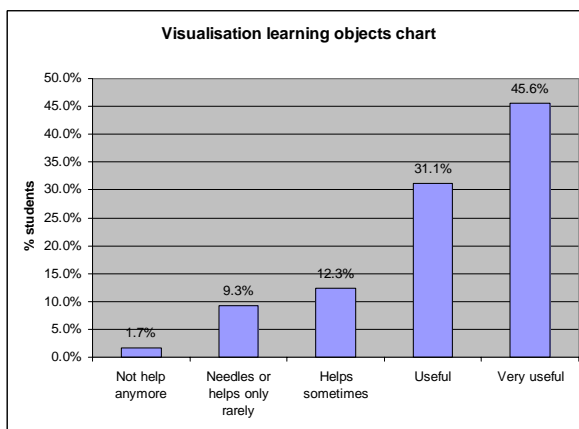


Figure 5

5. Conclusion

Interactive learning object is an idea that many teachers welcome in their search for new methods and support for novice programming students. What subjects to explain with the help of learning object is always a question and in the Codewitz project the need analysis was helpful for the project partners to choose where to begin. To

introduce the learning objects to teachers and students is still an ongoing process and all teachers can become partners in Codewitz, get access to the material bank and take part in developing more interactive learning objects.

It is quite clear that students believe that learning objects can be useful for them as novice programming students. But it is also quite clear that more introductions and better integration of learning objects is needed to encourage students to use them more frequently as a normal part of their programming study. Only a part of the students seem to use extensive material outside the classroom and although they know about good material they somehow do not use it. Here we might have to deal with students learning style and their immaturity as learners. Codewitz learning objects are not the witchcraft we might need in teaching programming today but I believe it could be useful especially if it becomes integrated into teaching and learning and a natural part of students programming life.

6. Acknowledgments

This work was supported by European Union's Socrates program Minerva: ODL and ICT in Education under "CodeWitz / Minerva - For Better Programming Skills" (109986-CP-1-2003-1-FI-MINERVA-MPP).

The authors thanks for assistance in the preparation of this paper to Codewitz Team, and to his son Ion Andrei (3 years old), for his understanding.

7. References

- [Ma 2005] Ma L., Ferguson J., Roper M. (2005). "A Collaborative Approach to Learning Programming: A Hybrid Learning Model", *Proceeding from the 6th Annual Conference of the Subject Centre for Information and Computer Sciences*. HE Academy. Ireland/UK.

- [McGill 1997] McGill, T. J. and Volet, S.E. (1997). “A Conceptual Framework for Analyzing Studetns’ Knowledge of Programming”, *Journal of Research on Computing in Education*, 29(3): 276-297.
- [McKeown 2004] McKeown J. (2004). “The use of a multimedia lesson to increase novice programmers’ understanding of programming array concepts”, *Journal of Computing Sciences in Colleges*, Volume 19 Issue 4.
- [Nugent 2005]Nugent G., Soh L., Samal A., Person S. and Lang J. (2005). “Design, Development, and Validation of a Learning Object for CS1”, *Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education*, Portugal.
- [Sun 2005] Sun S., Joy M. and Griffiths N. (2005). “A Innovative Use of Learning Objects and Learning Style in Pedagogic Agent Systems”, *Proceeding from the 6th Annual Conference of the Subject Centre for Information and Computer Sciences*. HE Academy. Ireland/UK.