USEFULNESS OF LEARNING OBJECTS IN COMPUTER SCIENCE LEARNING

The Codewitz project

Ásrún Matthíasdóttir
Assistant Professor
Reykjavik University
asrun@ru.is

ABSTRACT

In this paper the author explains the process in the Codewitz/Minerva project where the weight is on making interactive learning objects for teaching and learning programming. It is quite clear that students believe that learning objects can be useful for them as novice programming students. It is also quite clear that more introductions and better integration of learning objects is needed to make the students use them more as a daily part of their struggle with programming.

Key words: learning objects, learning and teaching programming, interactive multimedia, Codewitz/Minerva.

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. This question 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 fist language, other emphasize teaching methods and the structure 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 collaboration skills are among many desirable skills computer science students should be trained in (Ma, Ferguson, Roper 2005). Even the name of the course seems to matter, especially to female students (Asrun Matthiasdottir et al. 2004a).

Computer science students often have diverse educational backgrounds and learning styles that may 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 Some even state that, "…methods used to teach introductory computer programming to college students are becoming outdated" (McKeown 2004, p. 39). Research indicates that novice programmers have difficulty in understanding programming concepts, the syntax of codes and the interpretation of blocks of codes (McKeown J. McGill and Volet 1997).Teachers are looking for new methods and support for their teaching, they want to help students and motivate them and learning objects with their visualization may be considered a feasible support.

As Benset (2005) states the term learning object has been over-used and there are different definitions, strategies and standards of learning objects (Sun, Joy and Griffith 2005). IEEE has a broad definition of learning objects as "…any entity, digital or non-digital, which can be used, or reused referenced during technology supported learning." (Abernethy, Piegari, Reichgelt 2005, p. 12). Nugent et al. (2005) define a learning object simply as "… a structured, standalone media resource that encapsulates high quality information to facilitate learning and pedagogy" (p. 370).

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 participating universities in the Codewitz/Minerva project are the

University of Applied Sciences in Furtwangen in Germany (FHF), the Tampere University of Technology (TUT) in Finland, the Ventspils University College (VENTA) in Latvia, the Technical University of Civil Engineering (UTCB) in Bucharest, Romania and Reykjavik University (RU), Iceland.

Learning objects in the Codewitz project are web-based standalone visualisations of programming tasks or code examples (www.codewitz.net) built for clear specific learning goals. The Codewitz learning objects are so far manly 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 (www.codewitz.net) where plans 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.
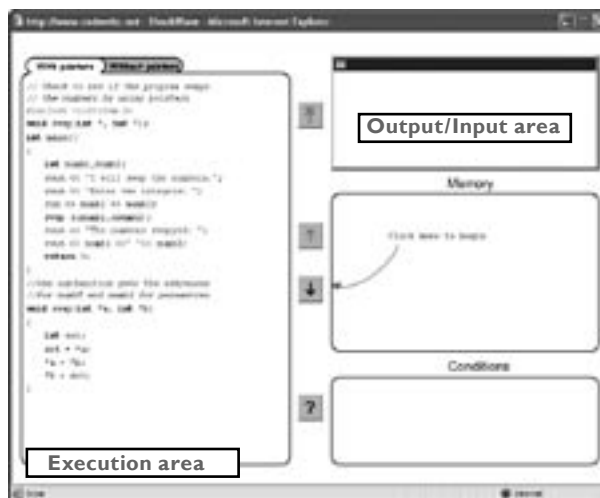
Figure 1

And in figure 2 we can see a slightly different design with an instructions console window.

Figure 2

In this paper the process of making the learning objects is explained and students' attitudes at Reykjavik University towards the objects are discussed.

TO PREPARE LEARNING OBJECTS

The Codewitz project started in 2003 with a need analysis where data was gathered by an online survey for students and teachers at the six participating universities to assist the group to choose what learning objects to make in the project. Among the main objectives was an emphasis on *"…recursion, language libraries, error handling, pointers and references. The need analysis also indicated that online interactive learning objects could be useful for students as they like to work on their own when learning programming."* (Asrun Matthiasdottir 2004b). The study revealed that students have different views on what is difficult to learn in programming and found the same things both difficult and easy e.g. dividing functionality into procedures, functions and/or classes and finding bugs in their own programmes. The teachers had different views from the students but agreed on some e.g. abstract data types and error handling (Asrun Matthiasdottir 2004c). Lahtinen, Ala-Mutka and Jävinen (2005) also discuss the results in their paper and state that *"…the most difficult concepts to learn are the ones that require understanding larger entities of the program instead of just details"* (p.17). The Codewitz learning objects are in a way focused on details or smaller parts of a program code but the code is shown in a proper program code to assist students to see the overall picture.

The need analysis was useful for the Codewitz group to start with and gave us guidance for choosing what subjects to emphasize when making plans for the learning objects. The next step in the project was for each partner to make up to 70 different plans where learning objects would be described and when this phase was finished in the autumn 2004 with over 400 plans it was time to evaluate them.

To evaluate the plans at Reykjavik University the author asked 31 computer science students (second and third year) in a course named *Information Society* to go to the Codewitz web, have a look at the plans and give their comments. The students gave 60 comments to the different plans they found on the web. The comments were very positive and in 20 of them it said that this was a clever web, useful for novice students and they wished they had had an access to it when they were first year students or as they said:

- *"Nice way to teach beginners to debug programs and help them to understand how functions connect and work together. This is also a good way for beginners to see how programs are constructed."*
- *"I think this is a very good page for people who are taking their first step in programming. This non-interactive exercise shows how a default error message can be used and that can be very helpful."*
- *"I like the use of visualization/animation in explaining step by step what is going on in the program as it runs. The text is simple and clear. I think this Learning Object gives a good instant feedback and could serve students well in understanding basic programming."*
- *"This is a useful way to show how fundamental features in programming work. This example is very clear and forward and shows what happens in every step. I like to see the comments of every step explaining what is going on."*
- *"Really good and fun exercises for beginners, and will probably help many people to get the basics of programming."*
- *"A good and simple way to teach and learn the first steps in programming. More educating than most books I have read. Also, this must be more fun, trying out simple examples which show what happens when the code is put to work. I hope there will be more of these examples for beginners be cause the first steps are often the most difficult."*

The comments from those who evaluated the plans were used to choose what plans needed to be updated or rethought and what plans to choose for further work. After the evaluation phase the work began on making the learning objects and each partner was supposed to make about 8 – 10 objects and put them on the project's web. To make the objects the partners could use different methods or programmes 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. 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.
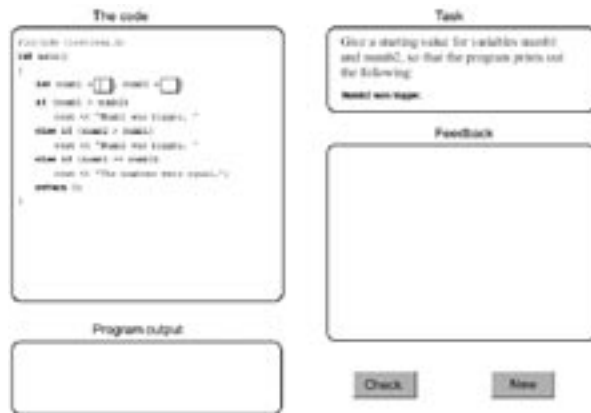
Figure 3

Now there are 178 learning objects in the Codewitz Material Bank (see http://www.codewitz.net/cmb.php) and to be able to use them your institution needs to be a partner in the Codewitz Network. You can join as a tester and a developer (level 1) or in addition to this, a Learning Object producer (level 2) (see http://www.codewitz.net/howtojoin.php).

USEFULNESS OF THE LEARNING OBJECTS

At the Reykjavik University we have not so far used the learning objects as an integrated part of the teaching programming; it has been more as a comprehensive (additional) material for the students to use when working on the course. The teacher have shown the students some learning objects, both in lectures and in workshops, and pointed out to them where to find them on the web.

To get some ideas about the students' attitudes toward the learning objects we put one question in an ongoing online web questionnaire for first year students in programming with C++. Of 33 students who answered a question about the usefulness of the learning objects 20 (61%) said that it was very useful or relatively useful in their study.

To get more useful information about the students' attitudes a question was sent by e-mail asking them about what they thought was positive and negative about the learning objects. Only 9 responded and three of them had not used the learning objects, only seen them explained in the classroom. One complained that he could not use the material on the web browser Opera because of the plug-in needed. The student wanted more information about what is needed and how to get the plug-ins if you are not using the web browser Internet Explorer. One student was extremely negative and said the material was rubbish. One student wanted more explanation of the tasks and one said that some of the objects needed to be simpler to understand. The positive comments were the following:

- *"Over all this was good and I used it a lot, especially the one where I was supposed to find the answer my self."*
- *"Codewitz exercisers are very good to get a grab of what is going on when you run a program, one can see how for example the loops are working and so on."*
- *"The main advantage is the understanding that the user gets for the function of each action in C++ and this step by step practical demonstration in the many examples and exercises gives a good support. The Codewitz learning objects are very useful as an interactive exercises and I can se no drawback."*
- *"What I saw seems to be clever but as I had god experience of b-programming before I came her I did not use this much."*

CONCLUSIONS

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 objects is always a question and in the Codewitz project the need analysis was helpful for the project partners to choose where to begin. Designing the plans was also good idea, you could plan what you wanted to do and get comments on your ideas before you started to make the object and there are now about 300 plans waiting for someone to turn them into real interactive objects. 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.

REFERENCES

- Asrun Matthiasdottir, HrafnLoftsson and Kolbrún Fanngeirsdóttir (2004a). Konur í Tölvunarfræði. Tímrit FUM.
- Asrun Matthiasdottir (2004b). *Learning objects in a multimedia interactive environment.* The Codewitz project. Proceedings of the International Conference on Computer Systems and Technologies (e-learning). Bulgaria.
- Asrun Matthiasdottir (2004c). *What students find difficult in learning programming.* Proceedings of 5th annual Conference of the LTSN Center for Information and Computer Sciences. Ireland.
- Benset, I (2005). *Toward Inclusive teaching Objects. Proceeding from the 6th Annual Conference of the Subject Centre for Information and Computer Sciences.* HE Academy. Ireland/UK.
- Lahtinen E., Ala-Mutka K. and Jävinen H. (2005). *Early programming: A study of the difficulties of novice programmers.* Proceeding of the 10th annual SIGCSE conference on Innovation and technology in computer science education ITiCSE '05. Portugal.
- 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, T. J. and Volet, S.E. (1197). A Conceptual Framework for Analyzing Studetns' Knowledge of Programming. Journal of Research on Computing in Education, 29(3): 276-297.
- 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 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 S., Joy M. and Griffithds 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
- Treu, K, Abernethy, K., Piegari ,G. and Reichgelt, H. (2005). A Learning Object Repository in Support Of Introduction to Information technology. Proceeding from the 6th Annual Conference of the Subject Centre for Information and Computer Sciences. HE Academy. Ireland/UK.