Providing Online Computer Science Programming Course Experience for Distance Learning Students

Tanja Krunić,
The Higher Education Technical School of Professional Studies, Novi Sad, Serbia

krunic@vtsns.edu.rs

Abstract

Distance learning is widely in use in many academic institutions around the world. Accordingly, many papers have been published focusing on problems distance learning students can have when attending online courses. Also, several guidelines have been published on how to organize distance learning courses in general. On the other hand, programming courses are specific, since the expected outcome of almost every programming course differs from simple reproducing of knowledge. In other words, after passing a specific programming course a student should be able to solve specific real world problems from that area. In addition, the rapid evolution of new technologies creates another set of problems for students. The idea of this paper is to derive a guide for organizing programming courses via distance learning. This paper focuses on a specific group of distance learners – the employed ones. In that manner, it points out the idea of encouraging employed distance learning attendees to solve work assignments for their study assignments rather than to create "toy programs". A guideline how to help distance learners to face new situations that occur due to the development of new technologies is given as well. There is also advice on how to predict the student's possible errors in code and how to help them avoid these errors in the future. Namely, predicting student's errors in code and warning them in advance can save them a lot of time, which is precious, especially for IT employed students.

Keywords: programming courses, distance learning, real world problems, toy programs, code errors

Introduction

Only two decades ago, the undergraduate student population was basically single, residential, and 18-23 years old (Beller & Or, 1998). Entering the information technology era, formal education has become essential for the economic success of individuals, organizations and countries. Under

these circumstances, the need of lifelong learning became an informal obligation for the modern employee. Nowadays, as the information technology era expanded employment skill sets, many employees have continuously to be retrained. For that reason, the number of older students has significantly increased. Moreover, these students are very often married, employed, and non-residential. According to the US Census

Press Releases (2006), 38% of all college students are 25 years and older. The majority of these older students (59%) attend school part time. Also, 21% of high-school students and about 50% of college students are employed.

Distance learning (DL) offers a good opportunity for employed learners. Formally, distance education is characterized by physical separation of learners from the information, an organized instructional program, use of technological media, and two-way communication (Heinich, Molenda, & Russell, 1993). One of the biggest benefits of DL is the issue of flexibility and time. DL students can complete their lessons whenever it suits their schedule (Distance Learning College Guide, 2009). Thus, the structure of DL gives adults a reasonable control over the time, place and pace of education.

So far, there have been several investigations about the barriers to DL in general. For example Galusha (2007) claims family obligations, the lack of face-to-face contact with the teacher, no support from the employer and isolation to some of the most common barriers to DL. On the other hand, it appears that many institutions of higher education are rushing into the new educational delivery systems without fully understanding how "place" and "time" factors impact student learning. Creating courses for distance learners without taking into account that they have specific problems in comparison to classroom learners discourages them in attending an online course, as can be seen in O'Malley and MacGraw (1999).

Attending programming courses on distance is specific. These courses are highly practical and require many hours of training. In addition, the rapid development of new versions of software requires students to be informed in this area continuously. Therefore, online programming course attendees meet additional problems in DL. Consequently, in section 'Electing appropriate study assignments', a small investigation was conducted with the aim to point out some of the problems of programming course attendees, especially the employed and married ones. The investigation involved questions like: Do employed student have enough free time for learning and completing assignments? Do they have the ability to obtain a free day to attend the classical exam? Do they spend too much time solving problems with code or newer versions of software due to the lack of face-to-face contact with the teacher? How can the teacher help students to overcome these problems?

So far, there have been recent theories in explaining the advantages of knowing the distance learner in general. Hughes (2004) suggests collecting information about the online learners before starting the learning process, such as the learner's readiness for online learning, access to and familiarity with the technology required, proficiency in the language of instruction, individual learning style, and educational goals, and how aspects of the individual's culture can affect learning. Milstein (2007) claims that knowing the learner helps creating an optimal curriculum. Similar, in the field of informatics, knowing characteristics of the learner is of great interest. Knowing some of the learner's characteristics could help the teacher in selecting an appropriate study assignment. What characteristics of the student are important to be collected prior to electing study assignments will be discussed in section "Electing appropriate study assignments."

The rapid software evolution can cause many problems in performing study assignments. Some commands might not be supported in newer versions of programs, and even the installation process or the complete environment can be completely different. Also, when completing programming assignments, students face code errors. Distance education systems now involve a high degree of interactivity between the teacher and student. They can communicate via e-mail, chat rooms, forums, etc. Moreover, virtual learning communities can be formed, in which students and researchers throughout the world who are part of the same class or study group can contact one another at any time of the day or night to share observations, information, and expertise with one another (VanderVen, 1994; Wolfe, 1994). On the other hand, finding the correct answer to stu-

dents' problems using forums, virtual communities and chat rooms requires extra free time for communication. Section 'Providing support in facing new situation and code errors' gives an alternative advice on how to provide support in case of code errors and encountering of new situations due to the development of software.

Nowadays, there are many DL systems in use around the globe. Even open source course management systems (CMS) that universities and community colleges use to add web technology to their courses, like Moddle (Modular Object-Oriented Dynamic Learning Environment http://www.moodle.org) are available free on the Web. According to Cole and Foster (2008) more than 30,000 educational organizations around the world use Moodle to deliver online courses and to supplement traditional face-to-face courses. But publishing the learning materials and tests on the DL system without providing additional information on how students perform tests and other important tasks, might be a complete failure. Section 'Important information about the course, technologies and performing tests' discusses what information, specific to programming courses, should be provided to the online learner.

Numerous studies have evaluated student achievement in specific DL programs in higher education. According to Abrami and Buras (1996), learning at a distance is seldom superior to traditional instruction, particularly for promoting higher level achievement and complex skills. In their research on student's perception on DL, O'Malley and McGraw (1999) found out that in general, students do not perceive that DL is as effective as traditional methodologies. The only claimed benefit of DL is that of working well with their schedules. This indicates that institutions and teachers have to make significant efforts in creating online courses. There have been many papers concerning strategies for creating effective DL courses in general. For example, Willis (1993) claims developing appropriate methods of feedback and reinforcement, optimizing content and pace, adapting to different student learning styles, using case studies and examples relevant to the target audience, being concise, supplementing courseware with print information, and personalizing instruction to be relevant factors to be concerned when creating DL courses. Similar suggestions for preparing learning materials can be found in Endean (2003). This paper, in its last section, provides a guideline for creating online computer science courses. The guideline gives instructions essential for creating programming courses that fit the employed distance learner.

Electing Appropriate Study Assignments

Finding an appropriate study assignment that suits every student is a complicated task for programming course teachers. In classroom teaching, creating "toy programs" is the only way for training code of a specific programming language. On the other hand, creating "toy programs" could be a waste of time for an employed distance learner, since there might be a couple of real world problems in his working organization suitable enough to be solved as a study assignment. Solving one of these problems would benefit the career of the distance learner besides training his programming skills. For example, a tourist agency employee who is retraining in the field of web design and/or web programming could create a database supported web application for the clients as a study assignment.

The Higher Education Technical School of Professional Studies, Novi Sad, Serbia, has a long tradition in the education of part-time employed students at weekends. The school is preparing for accreditation an e-business undergraduate program in distance learning. The part-time education is in some way similar to DL, since part-time students attend only a few lessons of every subject. During those few lessons, they become informed about the content of the course, get learning materials, and become familiar with the theory of the main topics of the subject. Unfortunately, there is little time for practicing. For the purpose of creating DL programming courses, a small survey involving 23 students was conducted. 22 of these students are part-time students, and one of them is an experimental DL student. For the purpose of accrediting the undergraduate e-business pro-

gram on distance, the teachers of each subject uploaded their learning materials on Moodle. The student attending the experimental program has access to appropriate subjects on the DL system.

The results of the survey are as follows. 57.9% of the students are undergraduate students, among whom 27.3% are studying web design, and 72.7% information technologies. 42.1% of the students are attending specialist studies in e-business. 94.7% of all students from the survey are full-time employed, among whom 63.8% in the field of informatics. 38.8% of the employed students spend on average 3 additional hours at work. 66.6% of the employed students spend additional time at home finishing their work assignments, due to short deadlines. Approximately half of them spend 2-3 hours for that purpose almost every day, while the other half claim that they finish work assignments at home only a few times a moth. This is a sign that there is not much free time left for other activities than work. Taking into account that 38.1% of the students from the survey are married, among whom 75% have on average 2 children, one can conclude that there is even less time for learning and doing study assignments. Namely, the average time available for learning and study assignments they claimed to be 2.3. All of the respondents of the survey agreed with the idea to do projects for their exams, rather than doing exams on the classical way. Their main argument for this is that the date and time of classical exams rarely suit their schedule. 78.9% of all respondents claimed they would appreciate if their study assignments were elected among some of their work assignments or projects. But among the students willing to do work assignments for their study, there were about 40% who claimed that they did not have the opportunity for that. These are mostly students who are employed out of the field of informatics. The reason for doing work assignments for their study was mostly claimed to be the lack of free time for creating "toy programs" (53.3%), and a chance to get a better practice (32.6%).

Therefore, it is advisable to publish a web form at the DL system at the beginning of the course in order to obtain information about the students. This questionnaire should collect information about the student's employment, working organization and workplace, programming skills, prior knowledge of technologies and programming languages the student already uses at work. A detailed list of unsolved problems in their working organization could further be specified during a teacher-to-learner chat session.

So far, some of the students of the specialist study group in e-business passed several exams through doing their work assignments. For example, one of the students, employed as a manager of a printing business, had a working assignment of creating a database supported user application for the business. She chose to use Microsoft Access for the database and C# for the application. This project was suitable to be a study assignment for subjects like databases and applied databases. In addition, the student created a website for the enterprise using HTML, Cascading Style Sheets and JavaScript, which was used as a project for the client-side coding subject.

Another important issue in ordering study assignments and projects are deadlines. Teachers should give reasonable project finishing deadlines for the distance learners, taking into account their employment and family obligation.

Providing Support in Facing New Situations and Code Errors

Students who attend DL programming courses can face many problems due to the lack of face-to-face contact with the teacher. Since DL students do not sit in the classroom with the teacher, the teacher cannot realize what the students' programming weaknesses are (i.e., what are their custom errors in code or in organizing the program). The teacher also does not know what problems students have in facing new situations.

The rapid evolution of software can cause many problems to DL students. Sometimes, some program commands can differ from version to version of the program. Imagine, for example, the teacher has provided students with the information how to install Internet Information Server on Windows XP. But the student has the latest computer at home working under Windows Vista. Unfortunately, the installation process differs from the one explained in the learning material. Thus the student might have problems he does not know how to solve alone. Unfortunately, as presented in the survey from the pervious section, employed students often do not have much free time at all. This means that they also do not have much free time for launching the internet and finding the correct answer.

Besides problems encountered from the appearance of new versions of technologies, there are also common mistakes occurring due to the student's inexperience. For example, a student creates a website and checks it in a web browser, say Internet Explorer. Being satisfied with its appearance, he/she decides to upload the website. After uploading the site and opening it in another web browser (Mozilla Firefox, Opera) or on a computer with different monitor resolution, nothing looks as desired, and the student remains completely confused. Such problems are encountered when student specifies the style sheets for only one browser or a fixed resolution.

To check if students' errors could be categorized, a small investigation was conducted on classroom lessons of a subject with topics on client side coding such as HTML, CSS and JavaScript, in the Higher Education Technical School of Professional Studies, Novi Sad, Serbia. The students were divided in two groups of approximately 30 students per group. During practicing exercises, the teacher noted the type and frequency of all errors made by students in the first group. Many of the errors seemed to be characteristic, since the frequency of several errors reached 25%. The other group of students was warned in advance not to make these errors, and it turned out that students from the second group completed their exercises approximately 30% faster than their colleagues from the first group. This indicates that informing students about common code errors can save them time in completing their exercises.

On the other hand, during the survey regarding part-time students mentioned in the pervious section some facts connected with code errors and problems encountered due to software evolution were collected, too. Namely, 78.9% of the respondents claimed they would appreciate if there were a list of frequently made code errors with the appropriate corrections available in the learning materials. Similarly, 84.2% of them would like to have such a list concerning problems due to the evolution of software. The majority of them 62.3% agreed that this kind of support would save them time significantly.

Taking into account all these facts, it could be helpful for the students if a programming course teacher collected common students' code errors and problems occurring due to the evolution of the software, and published them with the appropriate corrections and/or instructions in the DL system. Of course, the list should be updated from time to time. New generations of students may have new problems. This is a consequence of the increasing development of new versions of programs, web browsers etc. For example, new versions of web browsers support new commands, or do not support old commands of a script language.

The teacher willing to help learners with their code could also benefit from the publishing of lists with frequently made code errors and appropriate corrections, since this helps to avoid the receipt of dozens of emails with a big amount of "non-working" code.

Important Information about the Course, Technologies and Performing Tests

One of the most important things for DL in general, is to make the course attendees familiar with their duties and other important information about the course. Therefore, at the beginning of the course, there should be available easy to find documents providing information about the main course topics, the expected learning outcomes, how many credits it takes, what are the students obligations, projects, assignments, deadlines and in what form, when and where the exam should take place. Informing the students about all these important issues in advance is important for avoiding any kind of confusion. Clearly, the learning materials should be methodically organized, but this will not be discussed in this paper, since there are many guidelines for preparing online learning materials available at the Internet (for example, Endean, 2003).

A specific issue for programming courses is to provide students with additional information about the technologies to be used. It is not referable to afford students to use expensive licensed technologies that may be installed in the computer labs of the institution. Such technologies may be completely unavailable for a distance learner. It is complicated to provide access to the institutions computer labs for distance learners, since the provided access time may not match the distance learner's free time. Furthermore, the distance learner may not be residential, hence has no access to computers with the desired technology.

The best technologies for learning at a distance are free or low cost technologies that can be downloaded from the Internet. For example, a perfect programming language for an introduction programming course is Java. The Java runtime environment can easily be downloaded from the Internet and used without any limitations for free.

Another important issue in learning computer science subjects at distance is to provide students with enough information so that they can easily attend the course. This is especially important for tests. If tests are online, all instructions about how to perform important actions must be published on the DL system. For example, if students are supposed to draw algorithms in the test, and the used DL system is Moodle, there should be a short guide published in the test introduction, with instructions similar as suggested bellow:

Instructions for attaching your algorithm drawing to the test

•Use any drawing editor (e.g., Corel draw, Microsoft Visio, or even Microsoft Word or Open Office) to draw your algorithm.

•Save the appropriate drawing with the extensions .jpg or .gif. Maybe you will have to export your drawing for that purpose (if you use Corel Draw for example). Or if you use Microsoft Word you will have to copy your drawing and paste it into a program like Paint and then save it with the above indicated extension.

• Attach your drawing to the text editor given bellow this exercise. For attaching your drawing, just use the following icon:

If such instructions are not available in the DL system, it can happen that the student does not pass the test because he/she was just confused. In the above mentioned example, it can happen that the student has drawn the algorithm for the exercise quite well, but he/she does not know how to attach it to the test. While trying to find out how to attach the drawing, the time is out and

he/she does not do the task and, therefore, cannot complete other exercises that follow the one with the algorithm.

Guidelines for Organizing Programming Courses at a Distance

According to the above discusses issues on DL, the author of this paper suggests the following guidelines for organizing programming courses at distance:

•Upload a document with detailed explanations of the subject rules (the student's duties, information about tests, assignments, projects, exams, course credits, etc.)

•Upload a course topic overview.

•Upload a list of technologies (programs, browsers) that are supposed to be used for the purpose of the course. Attention: All of these technologies should be available for the student, and if possible they should be free or low-cost. If not so, allow students to use alternative technologies.

•Use tips for writing online learning materials. There are many of them available at the Internet (e.g., Endean, 2003).

•Upload all important information about actions to be performed while doing online tests.

•At the beginning of the course, provide an input form for the students with detailed questions about their workplace, work assignments and projects, knowledge of programs and technologies, etc. Alternatively, organize an online chat session with the student and discuss all these matters directly. An input form matches better if you have a lot of students, although you can achieve more detailed information through an online chat session.

•According to the data archived from the above mentioned input form, create, if possible, real world exercises (projects, assignments) for code training.

•Take into account if a distance learner is employed when setting deadlines for finishing study assignments.

•Collect the students' problems occurring due to the development of new versions of software and lack of literature. Upload an appropriate FAQ at the DL system.

•Upload a page with a list of frequently made code errors, and the appropriate corrections of the code, with brief explanations. The list should be updated from time to time.

Conclusions and Further Investigations

Several factors are influencing programming courses to be different from other, more theoretical subjects. For acquiring programming skills there is a need of completing many programming assignments and practicing. In that manner, electing an appropriate study assignment that fits the learner's knowledge needs, a small amount of free time, and desired programming tools, is a very important issue. Creating detailed profiles of every distance learner can help teachers to find an appropriate assignment. The desired information should include information about the student's workplace, work assignments and projects, programming experience and prior knowledge of specific technologies.

Publishing lists of frequently made code errors and problems encountered due to the evolution of software could help distance learners to complete their study assignments on time. Further inves-

tigations could include an analysis which would include two groups of distance learners. One group should be provided with a list of frequently made code errors and problems encountered due to the evolution of software, and the other not. It would also be interesting to perform a survey on how much time the teacher spend publishing these lists. The investigation could be focused on the comparison of the obtained results. The students from the group which would not be provided with any code error list could be encouraged to note their time spent on forums and chat rooms trying to find help. This data could further be used to verify the need of such a kind of support.

In addition, a similar investigation including programming course teachers could be performed, too. One group of teachers could provide lists of code errors to students and the other not. Both groups of teacher could note their time spent helping students with their code errors on forums, chat sessions and answering their emails. The obtained results of both groups of teachers should then be compared. Such a kind of investigation could provide us a measure on how much teachers benefit from the providing of DL students with code error lists.

References

Abrami, P., & Bures, E. (1996). Computer-supported collaborative learning and distance education. *The American Journal of Distance Education, 10*(2), 37-42.

Beller, M., & Or, E. (1998). The crossroads between lifelong learning and information technology: A challenge facing leading universities. *Journal of Computer-Mediated Communication*, *4*(2). Retrieved from http://jcmc.indiana.edu/vol4/issue2/beller.html

Cole, J., & Foster, H. (2008). *Using Moodle: Teaching with the popular open source management system.* O'Reilly Community Press.

Distance Learning College Guide. (2009). Retrieved from http://www.distance-learning-college-guide.com

Endean, M., (2003). *Learning materials at a distance*. The UK Centre for materials education.

Galusha, J. M. (2007). *Barriers to learning in distance education* University of Southern Mississippi. Retrieved from www.infrastruction.com/barriers.htm

Heinich, R., Molenda, M., & Russell, J. (1993). *Instructional media and the new technologies of instruction* (4th ed.). New York: Macmillan.

Hughes, J. A. (2004). Supporting the online learner. *Theory and practice of online learning* (chapter 15). Atabasca University. Retrieved from http://cde.athabascau.ca/online_book/ch15.html#one

Milstein, E. (2007). Meeting the needs of all learners. *Ludwigsburg-Beit Symposium.* Retrieved from http://www.ph-ludwigsburg.de/html/9e-aaax-s-01/seiten/SymposiumBB2/Milstein.ppt

O'Malley, J., & McGraw, H. (1999). Student's perceptions of distance learning, online learning and the traditional classroom. *Online Journal of Distance Learning Administration, II*(IV). State University Of West Georgia, Distance Education Center

US Census Press Releases. (2006). Available online at http://www.census.gov/Press-Release/www/releases/archives/facts_for_features_special_editions/007108.html

VanderVen, K. (1994). Viewpoint: The power and paradox of distance education. *The On-line Chronicle of Distance Education and Communication* [On-line journal] *7*(2). Available Usenet Newsgroup alt.education.distance, May 3, 1994.

Willis, B. (1993). *Strategies for teaching at a distance*. ERIC Document Reproduction Service No. ED 351 008

Wolfe, L. (1994). The digital co-op: Trends in the virtual community. *Paper presented at the Writers Retreat on Interactive Technology and Equipment. Vancouver, BC: The University of British Columbia Continuing Studies.*

Biography

Tanja Krunić is a lecturer at the Higher Technical School of Professional Education, Novi Sad, Serbia. She teaches thee programming courses: Introduction programming course, Client side programming for internet and Server side programming for Internet. She holds a MS in mathematics and is currently working towards her PhD in Mathematical Analysis from the Faculty of Mathematics and Natural Sciences, Novi Sad. Her research interests include important issues like usability, accessibility, privacy, and security on the World Wide Web. She is also doing researches in the field of Numerical Analysis.