e-Education in Teaching Programming - Forty Years of Promises?

Ljubomir Jerinić, Mirjana Ivanović, Zoran Putnik, Zoran Budimac, Miloš Savić

Abstract: *Instructional computer programs are being developed since the early '70s. Rapid development of Information Communication Technology, internet and mobile communication, introduction of computers into schools, and daily use of computers by people of different vocation, has made education a very important field to researchers.*

*From the beginning of the decade, Massive Open Online Courses (MOOC) has become a very popular form of additional education. The new technologies and methods like Video streaming, Flipped Classrooms, Cloud Computing, Mobile Computing and so on, have recently been penetrated into the modern e-Learning systems. These new technologies, techniques and methods inspired us to intensively use new term and way of integration of the computers in education called "**e-Education**". In this paper the state of art of e-Education in teaching and learning programming is briefly presented.*

Key words: *e-Learning, Computer Science, Teaching Programming, Pedagogy*

INTRODUCTION

The term "e-Education" we first time introduced in [9]. It refers to the application of Internet technology to the delivery of learning/teaching experiences. e-Education takes place in formal electronic classrooms, on corporate intranets used for just-in-time training, audio and video teleconferencing and in a variety of other technology-mediated learning spaces. The primary tools of e-Education [5] are e-Mail, e-Meetings, e-Expeditions, and the methodologies of a pedagogy known as e-Learning.

According to Huawei Enterprise Ltd. [6] IT-based education, or as we like to call it e-Education is undergoing rapid development. This virtual approach pointed that traditional school organizations are transforming into region-based, resource-sharing enterprises or "get-up-and-go" methodology. Centralization of teaching operations is expected to solve many of the problems associated to traditional education models, such as operations and actions inefficiency, duplicate or identical asset and investment, education resource segmentation, and decentralized administration. Huawei Enterprise Ltd. [6] approach anticipate high-quality education resources which are joined within a region, and educators could work together through comprehensive sharing, collaboration, and communications to make significant improvements to the quality of education services.

e-Education emerged from correspondence schools and audio/video courses. Literature, textbooks and articles recognize and often confuse several terms: Distance Education, on-line courses, on-line teaching, web-based courses, web learning, etc. Literature also recognizes limitations of e-Learning: "If something can be learned *exclusively* using software, without human communication, it's *not worth* teaching at the faculty; it's more suitable for some skill-acquiring courses." [7]

As a part of e-Education, e-Learning over the past 20 years has gone through a large expansion, both in research and commercially. Many of the research projects were carried out with the aim of popularizing, use and standardization of these new types of learning, and the use of computers in the educational process.

As part of the research one direction achieved significant results in particular, i.e. defining of technical standards for working communication between the applications and services of e-Learning. More specific, the development of the specification and adoption of appropriate standards that would allow the creators and users of various systems of electronic learning to work together and exchange material is of utmost importance.

One of the most important is IMS Global Learning Consortium (usually referred to as IMS GLC, IMS Global or simply IMS http://www.imsglobal.org/), a global, nonprofit,

member organization that strives to enable the growth and impact of learning technology in the education and corporate learning sectors worldwide. The main activity of IMS GLC is to develop interoperability standards and adoption practice standards for e-Learning, like Learning Tools Interoperability (LTI), Question & Test Interoperability/Accessible Portable Item Protocol (QTI/APIP), Common Cartridge, Learning Information Services and Content Packaging. In the rest of the paper, we will look back at our approach in comparison and assessed by the IMS Global Consortium approach.

What about teachers and about "new pedagogy" in terms of suitable teaching methods that could be used in e-Education? Are they prepared for the new challenges?

From the instructional design point of view, the e-Education could be defined:

"e-Education" = "Traditional teacher-led training" + "Materials of computer-based training".

Much has been promised about the potential of technology to revolutionise learning, with benefits identified in: Connectivity (access to information is available on a global scale), Flexibility (learning can be performed any time, any place), Interactivity (assessment of learning can be immediate and autonomous), Collaboration (use of Web 2.0 tools can support collaborative learning beyond the classroom), etc. Moreover, recent GALLUP poll[5] found that 33% of Americans believed that online education provided the best curriculum options and was a better value economically compared to traditional classroom education.

Thus, the time has come that university professors, and researchers in the field have to pay more attention about how the technology could be properly used. Therefore, the greater thoughtfulness and attention have to be given and specified on the part of e-Education, called "**e-Teaching**".

The rest of the paper is organized as follows. In section 2, we will try to identify some aspects and show the current state of e-Education. The third section is devoted to teaching programming in this context. The methods and techniques of e-Teaching are extended with pedagogical patterns, and their application in the teaching of programming is briefly described in the fourth section. The last section brings some concluding remarks.

"e-Education" = "e-Learning" + "e-Teaching"+"…"

According to Tavangarian et al. [13] e-Education[6] includes numerous types of media that deliver text, audio, pictures, animation, and streaming video, and includes technology applications and processes such as audio or video tape, satellite TV, CD-ROM, and computer-based learning, as well as local intranet/extranet, virtual classrooms and web-based learning. Current e-Education systems (which are predominantly e-Learning systems, because they lack the pedagogical aspects of teaching) on the market are limited to technical gadgets and organizational aspects of teaching, instead of supporting the learning. All of them through various tutorials provide only practical and technical support to users - teachers, ignoring and neglecting the need of pedagogy.

As a result, the learner, who learns through these e-Education systems, has become de-individualized. Possible solution is the creation of individual e-Learning materials and/or adaptive e-Education environments. For this purpose, a flexible

5 http://www.engineering.com/Education/EducationArticles/ArticleID/6767/GALLUP-Poll-Suggests-Online-Education-Has-the-Biggest-Bang-for-Your-Buck.aspx

6 To overcome problems with terminology, in this paper the term e-Education is used, regardless of whether considerations is related to a classical systems of e-Learning or a future systems for learning/teaching by computers which will include e-Teaching

multidimensional data model and the generation of individual content are possible solutions. Nowadays, term e-Education includes: multimedia learning, technology-enhanced learning (TEL), open education, life-long education, intelligent tutoring systems (ITS), computer-based instruction (CBI), virtual universities, computer-based training (CBT), computer-assisted instruction (CAI), web-based training (WBT), online education, virtual learning environments (VLE), m-learning, FLIP learning/classroom, Virtual Learning Environment/Course Management System (VLE/CMS), etc. These alternative names emphasize a particular approach, component, learning style, teaching and/or delivery method, but usually without enough emphasis on needed pedagogy.

Apart from these approaches, known and most used approach of a different type emerged from the theory of information systems. It is characteristic of current research in the field of e-Learning standards (IMS Learning Design standard, SCORM (http://www.adlnet.org/scorm/), etc.). The theory of these standards is trying to describe the teaching models in a consistent and computer-readable way. As a direct consequence of it, the very concept of the teaching model has a double meaning in this context:

- a set of machine-readable instructions describing the model, but also
- pedagogical support behind the deployment model.

For use in practice, in this approach tools for creating and tools for execution and the use of ready-made teaching models are being developed simultaneously. The results and achievements of the research, for example the IMS standard, can be found on the website of the project UNFOLD (http://www.unfoldproject.net).

On the other hand, A Massive Open Online Course - MOOC[7] is an online course aimed at unlimited participation and open access via web (http://www.thecompleteuniversityguide.co.uk/distance-learning/moocs-%28massive-open-online-courses%29/). In addition to traditional course materials, MOOCs provide interactive user forums that help build a community for students, professors, and other users. From the instructional design point of view, most of MOCSs are:

"e-Education" = "Traditional instructor-led training" + "Riches of computer-based training",

i.e. they represent blended learning models. Much has been promised about the potential of technology to revolutionise learning, with benefits identified in: connectivity, flexibility, interactivity, collaboration, extended opportunities and motivation. Nevertheless, from above consideration one could notice that the term "e-Teaching" is simply missing.

The next approach of future education could be

"e-Education" = "Collecting Information and Knowledge through Internet".

Transformed into a large collaborative learning environment, the Internet (from e-Education point of view) is comprised of information reservoirs, and according to Bell [1] it includes: (a) online classrooms, (b) social networks, and (c) virtual reality or simulated communities, to expeditiously create, reproduce, share, and deliver information into the hands of educators and students. Couros [4] extends Bell's definition and pointed that there are three major pools where individuals can acquire information:

(a) online classrooms including MOOCs,

(b) social networks including podcasts and video clips, and

7 MOOC; http://en.wikipedia.org/wiki/MOOCs_in_Europe

(c) virtual reality platforms, including 'Second Life' and 3-dimensional video games.

"e-Education" IN TEACHING PROGRAMMING

On the other hand, what is the situation in the field of e-Education in computer science, especially in the area of learning elementary programming? In this Section the current situation in application of e-Education in computer science, especially in learning elementary programming is concisely considered. In addition, the pedagogical approach in teaching programming, as well as suitable learning theories for learning programming is observed.

If you examine the contents of some popular e-Education systems for teaching programming:

- BubbleBee: Serbian site dedicated to the popularization of programming and its online education, sponsored by Microsoft, http://bee.bubblecup.org/,
- CODE: Every student have the opportunity to learn computer science, http://code.org/,
- CODECADEMY: Code interactively,

http://www.codecademy.com/#!/exercises/0,

- COURSERA: Take the world's best courses, online, for free,

https://www.coursera.org/

- CODE PROJECT: For those who code, http://www.codeproject.com/,
- KHAN ACADEMY: Changing education for the better,

https://www.khanacademy.org/,

you could find that the underlying educational philosophy is one of Objectivism [3]. This theory holds that the student's mind is an empty slate that the lecturer fills up. The systems approach to this kind of e-Education has the creator of that system examine the subject to be taught, divide it up into small bits, sequence the bits in some logical order, and then put all students through the same process of learning. From pedagogical point of view the creator use Semiotic Ladder [10] combined with Spiral [2] pedagogical patterns and/or teaching approach.

Almost all courses in programming from above e-Education organizations are focused and concentrated on LEARNIG SPECIFIC PROGRAMMING LANGUAGE instead of learning/teaching programming principles. Correspondingly, all the courses look like some programming languages manuals, with different examples - this is "e-Textbooks" approach, from pedagogical point of view.

For example, those "e-Textbooks" for learning elementary programming suggest that IF statements MUST come before LOOPING. The objectivist theory ignores the fact that such a methodology is deadly boring to the majority of students. It forces them to "learn" things they already know, and second, it ignores any individual difference in learning styles.

Constructivist educational philosophy according to Ernst von Glasersfeld [11], on the other hand, views the student as knowledgeable and task driven, where each learner should be treated as an intelligent, independently thinking individual. If every learner has his/her own way of processing learning material, there have to exist 'as many individual and unpredictable ways of learning as there are learners' [14].

Designing online learning material is a difficult task for novice teachers. Therefore, it seems that pedagogical patterns could help. Patterns are designed to capture best practice in a specific domain. Pedagogical patterns [2] try to capture expert knowledge

of the practice of teaching and learning. The intent is to capture the essence of the practice in a compact form that can be easily communicated to those who need that knowledge/experience.

In essence, a pattern solves a problem. This problem should be one that returns in different contexts. In teaching, we have many problems such as motivating students, choosing and sequencing appropriate materials and resources, evaluating and assessing students' achievements and results, and the similar. Each time a problem, pops up there are considerations that must be taken into account that influence our choice of solution.

A pedagogical pattern is supposed to present a problem and a solution. The problem together with the forces must apply to make that solution beneficial.

e-Teaching "OUR" PEDAGOGICAL PATTERN APPROACH

Designing Web-based content for e-Education for learning programming is a difficult task for novice teachers and results are poorly designed courses and learning contents. In [12] authors distinguish four methods in teaching programming from pedagogical point of view:

- The first approach, code analysis, consists of recognizing the parts of the artifact and/or parts of program and teacher try to explain what they mean are. It involves learners to analyze and understand existing code prior to producing their own.
- The second is the building blocks approach. The learner learns elementary constructions first, before constructing program.
- A third approach is identified as simple units in which learners' master solutions to small problems before applying the learnt logic to more complex problems.
- The forth approach, is called full systems approach. Learners design a solution to a non-trivial problem and the programming concepts and language constructs are introduced only when the solution to the problem requires their application.

A completely different way, pedagogical patterns approach, were proposed in [2] as an appropriate methodology for teaching programming. Designing pedagogical patterns help teachers to better organize the content and thus benefit the students who will use them. These patterns are developed for classic face-to-face teaching, mostly.

In our approach of teaching programming, we have used some of Bergin's pedagogical patterns, with extensions and modifications for e-Teaching and blended teaching, like *Concrete to Abstraction* [2]. We also define some new pedagogical patterns [8]: *Gradual Improvement*, *Stepped Development*, *Fine-tuning*, *Bed Cop Good Cop*, etc. The core idea is that these patterns could be used for teaching programming in e-Education, classical face-to-face, as well as in blended teaching. In the rest of this section the usage of pedagogical pattern *Wolf, Wolf, Mistake* [8] for learning by mistakes is presented.

Novice students make mistakes in programming, design, and particularly in problem solving. Students "believe" that teacher is a person who always tells the truth, so they accept the facts and solutions without checking them. Students often do not know how to interpret the error messages, or what to do to solve problems that are diagnosed. Debugging and Testing are an essential skill, whether done with a sophisticated debugger, or just by comparing actual outputs or results with expectations, as well as to have the whole picture of the problem and test properly the given solution from teacher.

Having all mentioned in mind, following learning scenario with pedagogical pattern *Wolf, Wolf, Mistake* could be performed:

- Some carefully chosen example in problem solving technique is presented to the students.
- Teacher creates solution from the beginning (understanding of the problem) to the end (making the code).
- The given solution has certain (hidden) specific errors (usually a single error).
- Teacher then asks students to explore given solution carefully, to test it, and to find is it good.
- When the students find the error, give them the chance to elaborate and discuss the cause and the consequences.
- If students do not find the error, tell them that the solution is not good in some cases. Give them extra time and/or some hints, trying to activate them.
- Repeat abovementioned process until the solution is found.

This method is very applicable to the early stages of learning programming. Syntax and semantic errors are frequent and students need to become familiar with the messages produced by compilers and run-time systems. Moreover, this pattern is good in teaching the students about importance of proper testing of the solution. The pattern could also be used in an analysis or design course in which certain errors could be made easily.

Pattern *Wolf, Wolf, Mistake* could be used effectively in teaching some introductory CS course. If you wish to teach the students about importance of analyzing the boundary cases in program design, and why the testing software is not an easy job, this pattern may be used. The pattern was used in some basic course for elementary programming and problem solving. Topic, which is taught, is divided into smaller pieces called subtopics or fragments. Fragments are introduced for systematic using *Spiral* [2] or *Semiotic Ladder* [10] patterns. The goal of the topic is to show usage of these fragments in solving certain problems. It is important that students must have active participation in constructing the solutions.

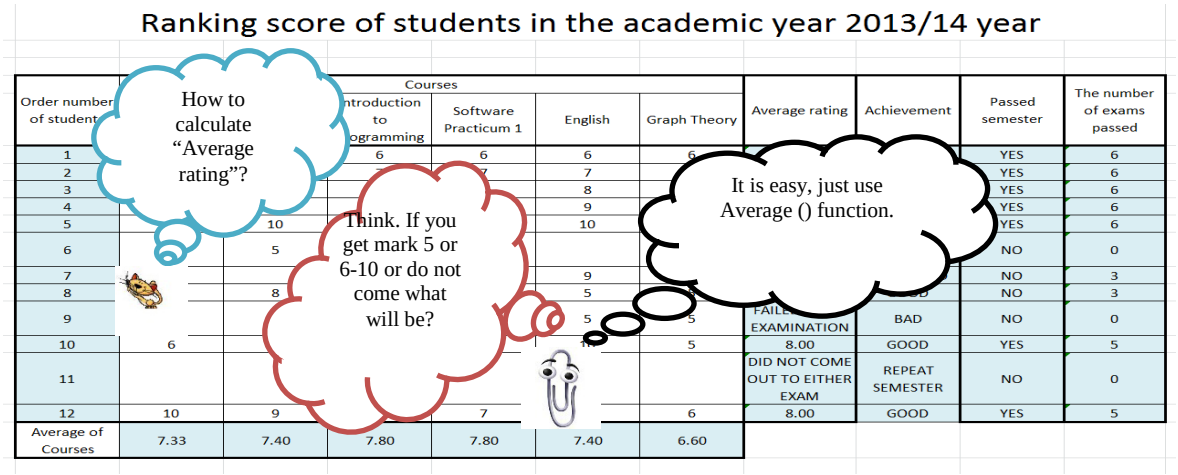

Figure 1 Example of usage e-Teaching pattern Wolf, Wolf, Mistake

Lecturer recapitulates data types and potential problems with them (such as division by zero for numbers) at the beginning of the class. Similarly, branching and control structures are summarized, and their usage in solving some problems is presented. The students together with lecturer solve some problem using these branching and control structures on specific data structures. The lecturer conducted the output. However, the "hidden" special case is not seen by students, i.e. for the particular

data entry the program could crash. They miss to observe the case, which leads to dividing by zero. This case lecturer "wisely" ignore in the analysis of the task. Next class, if students still did not notice the mistake, lecturer admitted his "sin", and explains the reason and consequences of mistake. If students ask question "How to solve the problem", lecturer could give right or wrong hint. (Figure 1.)

Couple of weeks later, students gets the assignment very similar to previously, but in some other context.

CONCLUSIONS AND FUTURE WORK

Today's world is rapidly changing, so teachers have to prepare students for a world that is totally unknown, for jobs that don't exist yet, and, hopefully, those future men/women will be able to both create those innovative jobs, and adjust to those created by others.

After years of work and many international projects concentrating on this, the basic problems appearing in connection with electronic learning is the impression that the majority of modern e-Learning system has significant deficiencies in at least one of the following aspects:

- poor pedagogical quality,
- deficiency of the portability, and
- lack of suitable tools for the development of both courses, and the very system of electronic learning.

Absence of pedagogical quality is considered the most significant, and it leads us to the necessity of introducing the notion of e-Teaching. To be successful, modern e-Education system must offer effective and attractive programs to students, based on approved pedagogical approach. At the same time, it has to provide a pleasant, easy and efficient working environment for the staff concerned with the development of teaching materials, curriculum planning process, providing technical assistance, and evaluation of results.

In this paper we do not deal with the problems of standardization, tools, interoperability, re-usability of existing e-Learning systems, notations and standards (IMS Learning Design standard, SCORM, etc.) for teaching programming. We suggest further development of the part of e-Education, the development of e-Teaching, with the application in teaching programming. The adaptation of existing Bergin et al. [2] and development of some new pedagogical patterns is presented. Also we exam specific results in patterns for e-Learning obtained by the E-LEN project (http://www2.tisip.no/E-LEN/index.php) and concluded that some of them have to be finalized and changed in order to be effectively used in teaching programming, as in classical F2F approach, as well as in e-Education, and of course in blended approach.

Technology alone does not improve education, but its integration in classroom could bring enough benefits to educational processes and activities. Students also should be integrated in this manipulation of technology and begin active participants.

Teachers need time to change their practices and share experiences with others in order to promote new teaching methodologies.

Programming courses are still important part of each computer science curricula. Investment in introducing innovative teaching approaches in programming courses that will motivate students to take more active role is very important.

Consequently, it seems that pedagogical patterns could be one promising methodology which will help teachers, especially novices, but also which will make educational environment more challenging for students.

We are planning in future work to develop variety of programming pedagogical patterns that will make introductory programming courses more attractive for students.

REFERENCES

[1] Bell, F. Connectivism: Its place in theory-informed research and innovation in technology-enabled learning. International Review of Research in Open and Distance Learning, 12(3). pp. 98-118. 2010.

[2] Bergin, J., J. Eckstein, M. L. Manns, H. Sharp, K. Maraquardt, J. Chandler, M. Sipos, M. Völter, E. Willingford. Pedagogical Patterns - The Pedagogical Patterns Project. Bergin J. (Ed.) Pedagogical Patterns: Advice for Educators. 2012.

[3] Bernstein, R. J. Beyond Objectivism and Relativism: Science, Hermeneutics, and Praxis. Philadelphia: University of Pennsylvania Press. Philadelphia. 1983.

[4] Couros, A. Open, connected, social - implications for educational design. Campus - Wide Information Systems, 26(3). pp. 232-239. 2009.

[5] Grimes, G. T., C. Whitmyer. e-Education: What is it? FutureU Press. 2009.

[6] Huawei Enterprise Ltd. Regional e-Education Solution. Web page at http://enterprise.huawei.com/en/solutions/trade/education/basic-education/hw-263530.htm

[7] Ivanović, M., Z. Putnik, Z. Budimac, Lj. Jerinić, B. Vesin. Some Aspects of e-Learning. Proc. of International Autumn School in Computational Intelligence and Information Technologies. pp. 14-16. 2004.

[8] Jerinić, Lj. Pedagogical Patterns for Learning Programming by Mistakes. Proc. of International Conference Computer Algebra and Dynamic Geometry Systems in Mathematical Education CADGME 2012, Novi Sad, pp. 56-61. 2012.

[9] Jerinić, Lj., M. Ivanović, Z. Budimac, Z. Putnik. Computer Based Education - Twenty Years of Promises, but... Proc. of International Autumn School in Computational Intelligence and Information Technologies. pp. 21-30. 2004.

[10] Kaasbøll, J. J. Exploring Didactic Models for Programming", Tapir, pp. 195-203, 1998.

[11] Pörksen, B. Abschied vom Absoluten: Gespräche zum Konstruktivismus. Heidelberg: Carl-Auer-Systeme Verlag. 2001.

[12] Selby, C. C. Four Approaches to Teaching Programming, Learning, Media and Technology: a doctoral research conference, London, 2011.

[13] Tavangarian, D., M. Leypold, K. Nölting, M. Röser. Is e-Learning the Solution for Individual Learning? Journal of e-Learning, Vol. 2, Issue 1. pp. 30-35, 2004.

[14] Thissen, F. Das Lernen neu erfinden - konstruktivistische Grundlagen einer Multimedia-Didaktik. In: Thissen, Frank: Material zum Thema Lernen. 1997.

ABOUT THE AUTHOR

Assistant Ljubomir Jerinić, M. Sc., Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Phone: +381 21 6350449, E-mail: jerinic@dmi.uns.ac.rs.

Full Prof. Mirjana Ivanović, Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Phone: +381 21 6350449, E-mail: mira@dmi.uns.ac.rs.

Assist. Prof. Zoran Putnik, Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Phone: +381 21 6350449, E-mail: putnik@dmi.uns.ac.rs.

Full Prof. Zoran Budimac, PhD, Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Phone: +381 21 6350449, E-mail: zjb@dmi.uns.ac.rs.

Assistant Miloš Savić, M. Sc., Department of Mathematics and Informatics, Faculty of Sciences, University of Novi Sad, Phone: +381 21 6350449, E-mail: svc@dmi.uns.ac.rs.

The paper has been reviewed.