The Effects of a Web-Supported and Well-Structured Problem-Based Method on Achievements of Learners

Murat Pasa Uysal
mpuysal@gmail.com. mpuysal@kho.edu.tr

Abstract

Different tools and methods have been proposed to enhance teaching computer programming (CP). One of them is the traditional teacher-directed instruction, which may be preferred by the majority of CP teachers or learners. However, learning CP remains difficult, especially for beginners. While Problem-Based Learning aims the self-direction and collaboration in learning, it is also believed to have improved CP skills. However, there may be some obstacles. Even the possibility of a change in teaching method may be frustrating for participants, let alone complete restructuring of a course or curriculum. Therefore, this paper summarizes the results of an empirical study on teaching CP through a blended method. The research purpose was to explore how a web-supported and well-structured problem-based method would affect the academic achievements of CP learners. The 433 freshman undergraduate students participated in the study and took the "Introduction to Computer Programming with C" course. They were randomly assigned to either of the experimental groups: web supported & well-structured problem based or web supported only. The Group-1's instruction was designed according to the "Instructional Design Model for Well-Structured Problems". A web-based tool supported the instruction not only by providing the instructional materials, but also by presenting the worked examples, conceptual diagrams and practice problems. The Group-2's instructional activities were similar to a traditional CP course, which directed by a teacher aligned with the contents. The web-based tool posted only the course's materials of the Group-2. As a result, we found a statistically significant difference between the study groups according to the academic achievement tests.

1. Introduction

Teaching and learning computer programming (CP) has been one of the most researched topics in educational computing. Many tools and techniques have been proposed to enhance understanding of programming. However, it remains difficult especially for beginners. The traditional teacher-directed instruction could be one of the important reasons, at which an instructor plays the role of primary source of knowledge and expert. Learning is assumed to occur if students understand the instructor's explanations and transfer them to the programming tasks. The main emphasis is put on successful application of knowledge to programming tasks during lab hours after presenting the contents in lectures. Although this approach may be preferred by the majority of CP teachers or learners, it is still away from preparing engineering students for the challenges of workplace problems [1]. Therefore, instructors should be in search of the ways for integrating innovative techniques to their classes to overcome the difficulties in learning or teaching programming.

Studies indicate that expert computer programmers consciously employ problem-solving strategies such as planning, designing, coding and testing, and they explicitly decompose problems into subproblems [2]. However, novice programmers generally attempt to start coding without approaching to the problem as a whole, planning the solution and applying solution steps automatically [3]. It is also known that low-performing students generally lack the skills and cognitive strategies i.e. reflecting on a programming problem, organizing information, and exploring different solutions. The generalization of CP skills is another issue for beginners. Novice learners tend to forget the details of programming techniques if they cannot meaningfully process the information. Therefore, additional instructional strategies can complement a CP course, and this might be more effective than teaching programming facts and rules only. For example; learning CP with problem solving techniques, such as the top-down design, or breakdown of a given problem, has already been used by instructors. However, integration of problem solving and CP skills is not easy, and it has to be grounded on instructional principles.

Problem-Based Learning (PBL) has been one of the instructional methods aiming to enhance both self-learning skills and acquisition of knowledge through self-direction and collaboration. Indeed, problem solving plays an important role for adapting ourselves to the environment as well as helping us to tackle the issues in daily life. On the other hand, CP skills are believed to have improved problem solving skills vice versa. Thus, both of the topics have attracted many researchers hoping to explore different variables within the framework of teaching CP. However, there may be some obstacles. In the first place, teachers and learners have to change their attitudes. Even the possibility of a change in teaching method may be frustrating for participants, let alone complete restructuring of a course or curriculum. Institutional culture, current evaluation criteria, allocation of more time and resources, and commitment to PBL are the important factors for the success of a problem-based instruction. Therefore, a blended approach or a gradual transition to a PBL environment might be suggested, and it would be less discouraging when regarding these potential problems. Furthermore, this type of hybrid approach would also provide valuable experiences for the design of problem-based instructions in the future. Therefore, this paper summarizes the results of an empirical study on teaching CP through such a blended method. The primary purpose is to explore how a web-supported and well-structured problem-based learning would affect the academic achievements of CP learners. The following sections present the background theory, research method, results and discussion parts of our paper.

2. Problem solving and computer programming

In general, a problem is defined by a problem domain, a problem type, a problem solving process, and a solution [8]. The problem domain consists of concepts, rules, and principles defining the problem elements. The problem solving process includes both the problem representation and understanding of the problem. It requires the activities with a set of operators in order to move from a problem state to a goal state. The problem type is determined by the combination of concepts, rules, and the procedures specific to the problem. CP and problem solving have much in common. In essence, programming can be regarded as a type of problem solving process. Both programming and problem solving require high order cognitive skills. They engage a variety of cognitive components, such as propositional information, concepts, rules and principles [8]. Both of these disciplines adopt a similar approach to problem solving. A problem solver/a programmer has to: (1st) identify and understand the problem/the programming task (2nd) search and find alternative ways to solve the problem/the programming task, (3rd) select the best solution/the algorithm, (4th) apply the solution/develop the code, and (5th) evaluate the solution/test & debug the code.
The well and ill-structured problems are the most encountered types, which can be used for the design of a CP course depending on learning outcomes. The ill-structured problems are similar to the ones in everyday practice or workplace problems. They are not constraint by the contents studied in classrooms. The situations in ill-structured problems are not well defined and clear. They may require integration of different knowledge domains, and their solutions are usually unpredictable. On the other hand, the well-structured problems are constraint to a certain situation requiring application of finite number of rules, principles and concepts. All of its attributes and components have to be presented to the solvers, and therefore a well-structured problem has a preferred or prescribed solution. This type of problem is naturally more common in schools [8]. Although the studies that compare the skills for a well or ill-structured problem make some distinctions, well-structured problem solving practices are believed to have positive effects on general problem solving performances. Moreover, the studies both in the software industry and in colleges imply that students have to internalize good programming skills long before they attempt to produce professional software solutions.

3. Method

The primary purpose of this study was to investigate the effects of a web supported and well-structured problem-based method on performances. The 433 freshman undergraduate students, who were 2.3% female (n=10) and 97.7% male (n=423), and aging from 18 to 20, participated in the study. They took the “Introduction to Computer Programming with C” course in 18 sections, and randomly assigned to either of two experimental groups; (Group-1): web supported and well-structured problem based; and (Group-2): web supported only. The Group-1 was administered as the treatment. The null hypothesis stated that no significant difference would exist between the study groups in terms of academic performances. The learners had no prior experience in CP. During the course, the fundamentals of programming concepts were given, and it consisted of a two-hour lecture and a one-hour lab per week, with 15 weeks per semester. At the end, an open-ended academic achievement test given to determine whether the students acquired required skills and knowledge. The experienced CP instructors evaluated the test for the content and face validity. The instructional environments were also supported by a web-based tool. However, the instructional approaches differed for both of the study groups as follows:

The Instructional Design for Group-1: The instruction was designed according to the Jonassen's [8] six-step "instructional design model for well-structured problems". Since the participants were first-time programmers, it was hoped that this approach would motivate the learners and help them to internalize basic programming skills and knowledge. Each weekly lecture included the first three steps of the proposed model, and the next steps were in the lab hours. The main focus was on the programming problems, the representations and solution processes of the programming tasks. The lectures and labs were integrated to model CP process as a form of well-structured problem solving process and its required activities. The web-based tool supported the instruction not only by providing the instructional materials, but also by presenting the worked examples, conceptual diagrams and practice problems. Each session started with a well-defined programming problem that required the execution of following steps:
Step-1: Reviewing concepts, rules and principles of CP: The concepts, principles and techniques related to the programming problem were initially reviewed at this step. The learners were informed about the basic skills and techniques required for problem solving using CP.
Step-2: Presenting conceptual models of problem domain: The CP facts and rules related with the programming problem were presented as the problem solving elements, and they were associated with the problem parts, attributes, and states. The diagrams and concept maps showed the interaction between the problem elements and programming constructs.
Step-3: Presenting worked examples for modeling problem solving performance: At this step, the problem-solving process using CP was modeled with worked examples similar to the sample problem. The learners reflected on the worked examples and their solutions, and learned how to construct the problem representation of a given programming task.
Step-4: Presenting practice problems: Although problem solving schemas may develop quickly in learners, the worked examples are not sufficient alone [8]. During this step, different practice problems aimed to facilitate programming schemas, and to transfer the CP knowledge to novel problems. The programming problems were made more realistic by withholding some programming elements, or including irrelevant variables in the programming tasks.
Step-5: Supporting the search for different solutions: It was important to provide learners with different problem solutions when helping to construct CP schemas. Therefore, a variety of strategies supported the search for different solutions, i.e. "recalling a previously developed program" or "breaking down a problem into sub-problems". These strategies assisted the learners in developing skills for generating different solutions and algorithms.
Step-6: Reflecting on problem state and problem solution: Learners had to reflect on the different problem states and the conditions. It was essential to note to the characteristics of the programming problems, their known and unknowns, and the situation in which they were stated [8]. During this step, the learners focused on the solution processes, which were especially effective or not effective in solving the practice problems. The students were expected to associate the programming problems with the successful solutions developed during labs or lectures.
The Instructional Design for Group-2: The activities were similar to a traditional CP course. The teacher directed the instruction according to the content during the lectures. The focus was on the programming language itself, syntax rules and principles. Students worked in the labs, and practiced on what had been presented in the lectures so far. The web-based tool posted only the course's power point presentations, assignments and handouts.

4. Findings and discussion

Based on the tests of normality, the academic achievement test was not normally distributed ($p< ,05$). Therefore, the Mann-Whitney test was used for the statistical analysis. As a result, a statistically significant difference was found in the academic achievement tests at the $\alpha = 0.05$ level of significance ($z = -3,211$; $p < ,05$). It is possible to state that the learners instructed with the web-supported problem-based method displayed higher academic performances (Table 1).

Table 1. The Mann-Whitney test results of the academic achievement tests

Groups	n	Mean rank	Sum of ranks	*z*	p
Group-1 (Web-supported & problem-based)	219	236,08	51701,50	-3,211	,001
Group-2 (Web-supported only)	214	197,47	42259,50		

Through the presentations of CP components in the form of problems, the learners were able to build domain-specific representations. The diagrams and concept models possibly helped them to develop meaningful CP schema. Presentation of the contents in the well-structured problem-based environment provided the structural knowledge at the level of appropriate detail and familiarity. The interaction maps of programming concepts and rules showed the interrelationships between the programming components, and made them explicit to the learners. The worked examples supported the novice learners and categorized the programming problems with similar solutions. By using the "means-ends analysis" and "problem decomposition" techniques, the learners were able to focus on the aspects of CP easily. This situation possibly made programming understandable, and enabled the learners to reach to the solutions with fewer steps. Integration of the worked examples and the practice problems helped automating the CP skills. The support for searching different solutions and the strategies, such as "breaking down a programming problem", assisted the learners in developing effective and efficient skills required for a programming task.

How information is coded and processed during CP instruction affects the quality and effectiveness of cognitive performance. The literature review suggests that learning CP with problem solving strategies: (a) improves general problem solving skills; (b) provides understanding of a language; (c) increases the skills needed for analysis, planning and interpretation; (d) enhances performances in other knowledge domains or disciplines [4-7]. Reflecting on programming facts and rules, which were in the form of problem states and solutions, possibly helped learners to develop stronger schemas. The learners were able to note the characteristics of a programming problem [8]. Furthermore, task automation and schema acquisition are already known to be effective in learning a complex skill, like CP [10]. This frees up the working memory, reduces the cognitive load, and information can be processed automatically without conscious effort [11]. Therefore, the well-structured problem based instruction is thought to have provided the learners with automated programming skills needed for high-level cognitive performances.

Finally Web, as delivery medium, supported the implementation of the well-structured problem-based instruction. It eased the presentation of a problem domain and learning contents. Different forms of hypertext and multimedia tools helped to develop meaningful programming and problem solving schema. The learners could easily review the CP concepts, principles and techniques using the web-based tool. The representation of programming knowledge in different forms, such as text, graphics and diagrams, helped presentation of worked examples. As the previous studies indicated, the web-supported instruction is also thought to have improved the students' information-searching ability essential for high-level problem-solving performances [12-14]. Consequently, the web-based environment provided an effective and efficient mechanism for the design and implementation of the problem-based instruction.

5. Conclusion

This experimental study identified whether a web-supported and well-structured problem-based instruction would lead to higher academic performances in CP. The students participated to the study in either a web supported problem-based or a web-supported only environment. The Group-1's instructional activities were planned and organized around the framework of the well-structured problem-based design model supported by a web-based tool. On the other hand, the same tool was used for posting only the course's materials of Group-2, in which the students learned CP with traditional teacher-directed method. As a result, we found that the learners instructed with well-structured problem-based method displayed higher academic performances.

The findings indicated that this method enabled the learners to frame their first-time programming experiences through a series of well-defined problem solving activities. The programming problems were used for acquiring CP knowledge rather than direct exposition to facts and syntax rules. However, the limitation of this study is that it cannot provide recommendations for ill-structured problems though these types are the most encountered ones. Different research variables need to be explored within the framework of web-supported problem-based learning. We believe that our research has showed the potential for using the web-supported problem solving as a means of improving CP skills.

References

[1] Jonassen D., Strobel, J. & Lee, C.B. (2006). Everyday problem solving in engineering: lessons for Engineering educators. Journal of Engineering Education, Vol. 95 (2), pp. 139-151.

[2] Jonassen, D.H., & Reeves, T. C. (1996). Learning with technology: Using computers as cognitive tools. In D.H. Jonassen (Ed.), Handbook of research for educational communications and technology pp. 693-719, New York: Macmillan, USA.

[3] Scruggs, T. E. and Mastropieri, M.A. (1996). Teacher perceptions of mainstreaming inclusion, 1958-1995: A research synthesis. Exceptional Children, 63, pp. 59-74.

[4] Lockard, J. (1986). Computer programming in schools: what should be taught. Computers in the Schools, USA.

[5] Pea, R. D., & Kurland, D. M., (1984). On the cognitive effects of learning computer programming. New Ideas in Psychology, Vol.2, No 2, pp. 137-167.

[6] Palumbo, D. B. (1990). Programming language/problem-solving research: A review of relevant issues. Review of Educational Research, Vol.60, No 1, pp. 65-89.

[7] Dale, N.B., Weems, C. & Headington, M.R. (1997). Programming and problem solving with C++. Jones and Bartlett, Sudburry, Mass., USA.

[8] Jonassen, D.H. (1997). Instructional design models for well-structured and ill-structured problem-solving learning outcomes. Educational Technology Research & Development, Vol.45, No-1, pp. 65-94.

[9] Sweller, J., Merriënboer, V., Jeroen, J.G., & Paas, F. (1998). Cognitive architecture and instructional design. Educational Psychology Review, Vol.10, pp. 251-295.

[10] Merriënboer, J.G., & Paas, G.W.C. (1990). Automation and schema acquisition in learning elementary computer programming: Implications for the design of practice. Computers in Human Behavior, Vol.6, pp. 273-289.

[11] Ericsson, K.A., Kinstch, W. (1995). Long-term working memory. Psychological Review, Vol.102, No. 2, pp. 211-245.

[12] Chiou, C. K., Hwang, G. J., & Tseng, J. C. R. (2009). An auto-scoring mechanism for evaluating problem-solving ability in a web-based learning environment. Computers & Education, Vol-53, No 2, pp. 261–272.

[13] Hwang, G. J., Chen, C. Y., Tsai, P. S., & Tsai, C. C. (2011). An expert system for improving web-based problem-solving ability of students. Expert Systems with Applications, Vol- 38, No 7, pp. 8664–8672.

[14] Kuo, F.R., Hwang, G.J. & Lee C.C. (2012). A hybrid approach to promoting students’ web-based problem solving competence and learning attitude. Computers & Education, 58, pp. 351-364.