An Innovative Approach to Improve the Teaching and Learning in Computer Programming by Using the Web Tutorial Applications

Mohd Norafizal Abd Aziz
Universiti Teknologi MARA Pahang, Malaysia
Email: mnorafizal@pahang.uitm.edu.my

Mahfudzah Othman
Universiti Teknologi MARA Perlis, Malaysia
Email: fudzah@pahang.uitm.edu.my

Zainab Othman
Universiti Teknologi MARA Melaka, Malaysia
Email: zainab_othman@melaka.uitm.edu.my

ABSTRACT

Computer programming courses are often claimed to be tough and, thus, high failure rates usually becomes an issue debated over these years. Taking this issue into account, several methods and techniques have been incorporated by the lecturers in teaching programming courses in order to improve the success rates. This paper discusses the factors that influence the success and failure in programming courses, as well as the techniques and methods that have been already introduced by previous studies. This paper also discusses students' preferences and materials that they have been using to facilitate their learning. Above all, the main intention of this study is to propose the use of web-based learning as an alternative tool that can be used in the teaching and learning programming. Random sampling was used in this study and the respondents were students who enrolled in the Diploma in Computer Science programme in one of the public universities in Malaysia. The data collected were analysed using the SPSS software to determine the significance of using web-based tutorials in programming courses. The benefits of employing web-

ISSN 0128-5599

based learning in classes as result from research findings is discussed in the paper. It is hoped that lecturers will be aware of the effectiveness of this application as it can help to improve their teaching and learning environment in programming classes.

Keywords: *computer programming, web-based learning, web tutorial*

Introduction

Computer programming is claimed to be a challenging intellectual task. Blackwell et al. (2001) mentioned that computer programming involves complex activities equivalent to other design and engineering activities. This claim is also agreed by other researchers. For instance, Johanyák et al. (2008) stated that programming is a complex and difficult task, thus, software engineers need to possess strong knowledge of programming languages and the development tools. A recent study by A. Aziz et al. (2009) at Universiti Teknologi MARA Pahang also showed a significant number of Computer Science students who considered programming courses as tough.

Due to the nature of the complexity and toughness in learning programming, there are solid evidence showing high failure rates in computer programming courses as claimed by Lahtinen (2005). In addition, Roddan (2002) reported that almost 42.2% of the students failed in the programming module and do not pass on to the second year in Computing Science course at the University of Glasgow. He also added that there was approximately 40% of dropout rate in 2000-2001. Guibert and Girard (2003) also mentioned high failure rates, which are between 25% and 80% worldwide in programming introductory courses.

Despite the fact that programming is a tough course and it has contributed to high failure rates worldwide, previous studies so highlighted the reasons of these high failure rates. One of the reasons cited is students often had to struggle to assimilate the concepts involved in the early stages of learning programming and eventually they will either withdraw or failing in the course (Maltby & Whittle, 2000). In addition, White, et al. (2002) who studied the cognitive differences among the students found that cognitive ability would give significant impacts on students' performances in the programming subjects. They also cliamed that cognitive differences between the different types of programming languages can also contribute to the factors of failing or succeeding in programming courses.

Other studies show varieties of causes such as 1) the difficulties among the novices to learn programming in a traditional way (Guibert & Girard, 2003), 2) difficulties related to specific language structure (Sajaniemi, 2002), 3) the nature of the programming subject itself where students do not have enough of resources and they suffer from lack of personal instruction (Lahtinen et al., 2005); and 4) the student groups are large and heterogeneous, thus, making it difficult to design the instruction that can be beneficial for everyone.

Problem Statement

For many years, researchers have been striving to find the right solutions to the problem. Reducing the numbers of failures in programming courses has become their main objectives. Some research include investigating and comparing the differences in programming languages, paradigms and environments to evaluate the ease of use and appropriateness of the programming languages (Mancy et al., 2004). Others include concentrating on applying educational strategies in teaching programming such as the use of concept maps and other visual instructional techniques and tools as proposed by Denton et al. (2005). Woszczynski et al. (2005) also studied the students' personality profiles as the predictor of success in computer programming. Other efforts involve the introduction of cooperative learning or pair programming as proposed by McDowell et al. (2003), Pollock et al. (2001) and Lasdon (1998). All of these studies have shown the effort to improve, introduce new ways or even create innovative approach in teaching and learning of programming.

Recent studies have shown new efforts in introducing the web as the mediator to learn programming languages. Lahtinen (2005) in her study came out with an interesting idea which was developing web-based visualizations of programming concepts for use in classrooms and for supporting independent learning. The project was named Codewitz (www.codewitz.net); a project that aimed at developing solutions that would benefit teaching and learning programming. In addition, Cheung (2006) also adopted the benefits of using the web to deliver lectures of object-oriented programming to promote the active learning beyond the classroom walls. He claimed that by using the web-based learning system, it has improved the teacher-student interaction, could provide analysis on students' participation, and has helped students with their compilation activities in real time. Another web-based system similar to Cheung's

(2006) prototype is developed by Chen et al. (2006) called a van Hiele Web-Based Learning System which also incorporated the concept of knowledge management. This web-based system includes email, discussion board, Internet assignment unit, tutorial unit, quick-run unit, expert template, and knowledge management unit to promote collaborative programming learning.

Web-based learning for programming courses has also been introduced in many higher education institutions in Malaysia. For example, Universiti Teknologi MARA has been using i-Learn portal to accompany the teaching and learning for varieties of courses including programming courses (http://i-learn.uitm.edu.my/index.php). Many lecturers are using this portal to disseminate lecture materials to students and students can even perform online programming quizzes and lecturers' evaluation via this portal. From this, online collaboration can be carried out and sharing of materials and knowledge between the groups of lecturers who are teaching the same programming courses can also be promoted, thus saving time, energy and resources.

However, despite the fact that there are many portals or e-learning websites developed , the impact of using the web learning concepts to the students' performances still remains elusive. For further investigation on how the web can be used as the medium to teach programming and how significant is the effect of online learning to the students' performances in programming courses should be carried out Therefore, the aim of this study is to introduce the concept of web-based learning in programming courses and also study the impact in determining its benefits to students in terms of their achievements in programming courses.

Web-Based Learning

These days, learning from the web is most demanding and appealing. Web-based learning is now becoming very much popular for all kinds of learning environments (Amin et al., 2007). This new phenomenon is encouraged by the fact that there is a rapid growth of information and communication technologies where it consequently promoting the learning through the World Wide Web.

Learning environment can be described as a place where learners may work together and support each other as they use a variety of tools and information resources in their pursuit of learning goals and problem-solving activities (Wilson, 1995). Whereas, web-based learning can be

defined as a hypermedia-based program that uses the attributes and resources of the World Wide Web to facilitate learning activities (Khan, 1997). It can also be considered as the place where learners and teachers interact (Teles, 1993).

The concept of web-based learning is allowing learners to learn by themselves where learners can be physically separated from teachers and from each other, and they can participate in the learning environment at their convenience (Chiang et al., 2005; Pantel, 1997). Furthermore, Santally et al. (2005) added that one of the advantages of delivering web-based educational materials is that the same content is delivered to a number of students and can be accessed with no restrictions of time and place.

Daramola et al. (n.d.) also added that the overcrowded classes and shortage of staff that can affect the standard of academic levels will also encourage the growth of potentials for the adoption of the web-based learning which has proved relevant to both distant and on-site learning. Dramola et al. introduced the integrated framework for web-based learning which is made up of four components which includes an on-line classroom component, on-line result checking component, on-line course assessment component and on-line course registration component.

Materials & Methods

The research method for this study will be divided into three parts as explained below:

1. Phase 1: Investigations

In order to investigate the materials that the students have been using to accompany their learning in programming courses, a set of questionnaires were given to a sample group consists of 80 students from the first year to the final year students. The questionnaires contain questions on the effectiveness of learning and teaching programming in a traditional way or using the web tutorials. From the analysis, we can see the percentage of the usage of web tutorials among the students.

2. Phase 2: Analysis and Findings

Both groups consist of 23 students for each class and are being taught by the same lecturer with more than 7 years of experience in teaching programming courses. Later, in both groups, students' performances in examinations will be recorded and then being compared to see whether the use of web-tutorials has significant impact to the students' achievements in programming classes. For us to see the correlations between Web tutorials usage and students' performance in programming language course, procedure of Bivariate Correlations using Spearman's Rho model is being used to measure the strength of association between these two variables. Another test that is conducted is t-test to further analyze the significant impact of using web tutorials in programming classes to students' performances.

Results & Discussion

The first part of this study analyzes the kind of materials that the students prefer to use to accompany their learning activities in programming courses. Table 1 shows the results for materials that have helped the students in learning programming. It shows that, the results are fairly even between all materials. However, the highest percentage is 43.8% where the students prefer to rely on the lecture notes or copies of transparencies from the lectures. The result also shows about 31.2% of the students agreed that using web-based tutorials or web applications have helped them in improving their learning of programming languages.

As mentioned earlier, two groups of students, where each group consists of 23 students from the final year Diploma in Computer Science

Table 1: Materials that have Helped the Students to Learn Programming

Factors	Percentage
Programming course book	33.8%
Lecture notes/copies of transparencies	43.8%
Exercise questions and answers	41.2%
Example programs given by lecturers	32.5%
Still pictures of programming structures	42.5%
Web based tutorials/applications	31.2%

studies were being selected. The test group is being exposed to Web Tutorial in Programming Paradigm class where the lecture notes, examples, exercises and tutorials are available via the application. Whereas, the control group is being taught using traditional method, where the lecturer will be teaching using white board and papers, and examples were given to them in the class without having them to download it via the web tutorial application. Before we elaborate more about the impact of the web tutorials usage to the students' performance, we first take a look at Table 2 to see the frequency of students' achievements in their final examination results for Programming Paradigm course.

Table 2: Students' Results in Programming Paradigm Course According to Sample Groups

Final Exam Result	Sample groups		
	Test Group	Control Group	Total
A	1	0	1
A-	1	1	2
B+	3	4	7
B	14	4	18
B-	3	4	7
C+	1	6	7
C	0	1	1
D+	0	1	1
F	0	2	2
Total	23	23	46

From Table 2, we can see that in the test group, there were higher achievements in students' results and there were no students who failed the course compared to the control group where there were few students who have scored D+ and F for the Programming Paradigm course.

Table 3 shows that the correlation is significant at the 0.05 level (2-tailed). The analysis shows that there is a quite strong correlation between the web tutorials and students' performance with $r = 0.367$, at the rate of 13.5% and a significant value of 0.012. For further analysis, a t-test had also being done to find the significant impact of web tutorials usage to the students' performance. From the analysis, it shows the significant value is 0.01, which is less than 0.05 (sig. 2-tailed). Therefore, we can conclude that the use of web tutorial applications can give significant impacts to the students' performance in programming courses.

Table 3: Correlations between Web Tutorial and Students' Performance in Programming Paradigm Course

			Web Tutorial	Students' performance
Spearman's rho	Web Tutorial	Correlation Coefficient	1.000	.367*
		Sig. (2-tailed)	.	.012
		N	46	46
	Students' performance	Correlation Coefficient	.367*	1.000
		Sig. (2-tailed)	.012	.
		N	46	46

Future Development and Use of ProPWeb Tutorial

Based on our finding of important use of web tutorial in computer programming as stated above, the use of web programming tutorial could be enhanced with the use of web-based application. The development of ProPWeb Tutorial is an alternative to improve the teaching and learning for students and lecturers. With this web-based framework, we believe it will help to improve the student performances in learning programming languages and also to improve the skills of programming by learning the skills not only using the manual or books but also with the use of web tutorial as part of their materials in studies the programming languages. The conceptual framework of ProPWeb Tutorial development is described in Figure 1 below.

ProPWeb Tutorial is built using the web-based applications. Figure 1 shows the conceptual framework of ProPWeb Tutorial. The conceptual framework shows three different components in the ProPWeb Tutorial. The first component is the online registration where it caters two different users, the lecturers and students. The application will set the level of users according to user's category.

The second component is the online classroom. In this component, the course content will be provided to the users. Both types of users can participate in the learning activities provided online. Group discussions and dialogues between the students and the lecturers as well as their classmates can also being done via emails and forums. The last component is the course assessment. In this component, students will be given few types of assessments such as assignments, quizzes and tests.

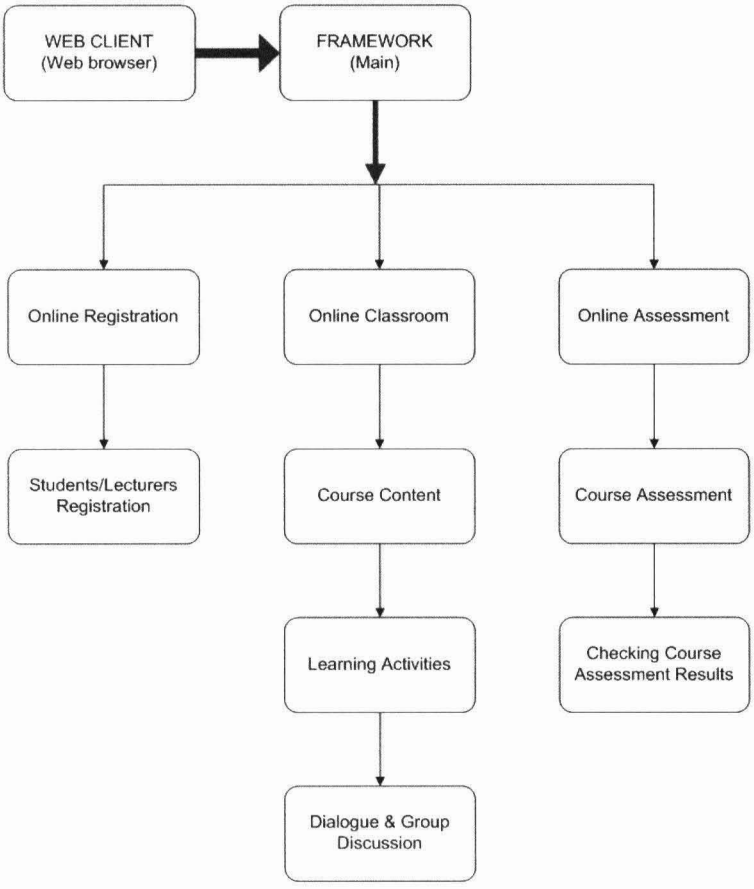

Figure 1: The Conceptual Framework of ProPWeb Tutorial

Lecturers will be able to check the marks and then provide the results. After that, students will be able to check their scores and grades via the Web.

Conclusion

There are various reasons why programming is considered hard to most of the students. Because of these reasons that have contributed to the high failure rates, many educators have tried to innovate their teaching and learning in programming courses. Various efforts have been

implemented and introduced whether in-class practices or the use of information and communication technologies which are widely used nowadays and more viable.

The results derived from Part 1 of this study have also revealed the fact that students have been using different materials other than lecture notes given in classes to accompany their studying and learning programming. One of the materials that most of them agreed have played an important role in their learning about programming languages is the web-based tutorials or applications. The high percentages of students who have been using web-based applications in their studies have shown that the Internet and web applications have significant impacts to the students' learning processes. Most of them agreed that by surfing the web applications, it has helped to increase their knowledge, improve their skills and understanding in programming.

From the results, it has been proved that Web-based tutorial application can be seen as another alternative that can be implemented in programming classes. Besides its viability, the classes would be more interactive. Resources can be accessed easier and knowledge sharing can be done instantly. All of these advantages can be seen as the motivation to improve students' performance in programming courses. Therefore, it is hoped that based on the findings of this study, academicians will find alternative ways in teaching and learning environment in programming classes by using the web tutorials as part of their lectures.

References

A.Aziz, M.N., Othman, M. & Othman, Z. (2009). The use of Web tutorial applications to improve students' performance in computer programming language. *Proceedings of International Conference on Languages 2009*, 535-546, Malaysia: Pulau Pinang.

Amin, M.R. & Anwar, S.S. (2007). An approach of developing of web-based learning environment by questioning and answering. *Asian Journal of Information Technology*, 6(7), 793-800.

Blackwell, A.F., Whitley, K.N., Good, J. & Petre, M. (2001). Cognitive Factors in Programming with Diagrams. *Artificial Intelligence Review*, 15(1), 95-114.

Chen, J.W. & Chih-Cheng Lin (2006). A van Hiele Web-based Learning System with Knowledge Management for Teaching Programming. *Proceedings of the Sixth IEEE International Conference on Advanced Learning Technologies*, 114-116, USA: Washington D.C.

Cheung, R. (2006). A web-based learning environment for object-oriented programming. *International Journal of Information and Operations Management Education 2006*, 1(2), – 157.

Chiang, H.C., Yang, C.C. & Chu, Y.P. (2005). Collaborative learning under an adaptive web-based architecture. *Information Technology Journal*, 4(4), 335-341.

Daramola, J.O. & Bamigbola, O.M (n.d). *An Integrated Framework for Web-based Learning*. Retrieved June 8, 2009 from www.unilorin.edu.ng/unilorin/publications/bamgbola/webframework2%5B1%5D.doc

Denton, L.F., McKinney, D. & Doran, M.V. (2005). A melding of educational strategies to enhance the introductory programming course. *Proceedings of the 35th Annual Conference ASEE/IEEE Frontiers in Education*, 7-12, USA: Indianapolis.

Guibert, N. & Girard, P. (2003). Teaching and Learning Programming by Example System. *International Symposium on End User Development*. Retrieved December 11, 2008 from http://www.uni-paderborn.de/cs/ag-engels/Conferences/SE-EUD/Guibert.pdf.

Johanyák, Zs. Cs., Pap-Szigeti, R. & Alvarez Gil, R. P. (2008). Analyzing students' programming failure, *A GAMF Közleményei, Kecskemét, XXII*. 115-120.

Kaasboll, J. (2002). *Learning Programming*, University of Oslo.

Khan, B.H. (1997). *Web-Based Instruction*. Educational Technology Publications, Englewood Cliffs, NJ, USA.

Lahtinen. E., Ala-Mutka, K., Jarvinen, H. (2005). A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin*, 37(3), 13-14.

Lasdon, L. (1998). The teachers' forum: Teaching non-linear programming using cooperative active learning. *INTERFACES*, 28(4), 119-132.

Maltby, J. R. & Whittle, J. (2000). Learning Programming Online: Student Perceptions and Performance. *Proceedings of the ASCILITE 2000 Conference*. Retrieved December 11, 2008 from http://www.ascilite.org.au/conferences/coffs00/papers/john_maltby.pdf/

Mancy, R. & Reid, N. (2004). Aspects of cognitive styles and programming. *Proceedings of the 16th Workshop of the Psychology of Programming Interest Group (PPIG 2004)*, 1-9, Carlow: Ireland.

McDowell, C., Werner, L., Bullock, H.E. & Fernald, J. (2003). The impact of pair programming on student performance, perception and persistence. *Proceedings of the 25th International Conference on Software Engineering*, 602-607, Portland: Oregon.

Pantel, C. (1997). *A Framework for Comparing Web-Based Learning Environments*. Postgraduate project report. University of Manitoba. Retrieved June 12, 2009 from ftp://ftp.fas.sfu.ca/pub/cs/TH/1997/ChristianPantelMSc.pdf

Pollock, L. & Jochen, M. (2001). Making parallel programming accessible to inexperienced programmers through cooperative learning. *ACM SIGCSE Bulletin*, 33(1), 224-228.

Roddan, M. (2002). *The Determinants of Student Failure and Attrition in First Year Computing Science*. Undergraduate project report. University of Glasgow. Retrieved November 21, 2008 from www.psy.gla.ac.uk/~steve/localed/roddanpsy.pdf.

Sajaniemi, J. (2002). Visualizing roles of variables to novice programmers. *Proceedings of the 14th Workshop of the Psychology of Programming Interest Group (PPIG 2002)*, 111-127, London: Brunel University.

Santally, M. Isaack & Senteni, A. (2005). Adaptation Models for personalization in web based learning environments. *Malaysian Online Journal of Instructional Technology (MOJIT)*, 2(1), 12p.

Retrieved June 8, 2009 from http://pppjj.usm.my/mojit/articles/pdf/April05/01-Santally-revised-typeset.pdf.

Teles, Lucio (1993). Cognitive apprenticeship on global networks. *Global Networks: Computers & International Communication, MIT Press*. Cambridge: USA, 271-281.

White, G.L. & Sivitanides, M.P. (2002). A Theory of the Relationships between Cognitive Requirements of Computer Programming Languages and Programmer's Cognitive Characteristics. *Journal of Information Systems Education*, 13(1), 59-66.

Wilson, B.G. (1995). Metaphors for instruction: Why we talk about learning environments. *Educational Technology*, 35(5), 25-30, 1995.

Woszczynski, A.B, Guthrie, T.C. & Sherri, S. (2005). Personality and Programming. *Journal of Information Systems Education*, 16(3), 293-300.