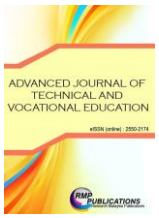
Advanced Journal of Technical and Vocational Education, 2 (3): 40-43, 2018
e-ISSN: 2550-2174

DOI: 10.26666/rmp.ajtve.2018.3.7



Student's Difficulties in Learning Programming

Hawariyah bt Rahim[1], Halimah Badioze Zaman[2], Azlina Ahmad[3], Nazlena Mohamad Ali[4]
[1]Jabatan Teknologi Maklumat dan Komunikasi, Politeknik Tuanku Syed Sirajuddin, Perlis, Malaysia
[2,3,4] Institute of Visual Informatics, UKM Malaysia

Abstract: Programming is a cognitively challenging subject. Hence, good instructional strategies and tools are important in providing the student optimal guides and learning support. Interviews with experts were conducted to identify major problems in the teaching and learning programming. The quantitative finding identifies that problem-solving is a major problem experienced when teaching and learning programming language. This paper describes how problem-solving skill can be developed by a viable mental model to help students in learning programming. In addition, the qualitative findings provide information about the current methods used to teach programming language and recommend visualization programs and games as learning tools to improve the current learning and teaching methods. Furthermore, this paper proposed a solution based on game and constructivist approach as a tool to improve students' mental model in learning programming.

Key words: *Programming difficulties, problem solving, mental models, constructivist approach, games.*

INTRODUCTION

Programming language is a subject offered in the field of Information Technology and Computer Science education. The main objective of this course is to create graduates who are highly skilled and competent in the field of programming. Recently, there are reports saying that there is a reduction of more than 50 percent of enrollment in Computer Science students in universities. Failure rate and high dropout among students of Computer Science in programming has attracted attention of researchers to investigate the cause of the problem. Students found difficulties to solve programming problems in the form of program codes from the basic level, even though they have no problem to write and explain it verbally. This problem occurred because the students failed to write a good algorithm. This skill is one of the major components in problem solving phase in programming [1][2]. To write a good algorithm and solve the problem, a student needs to think analytically [3][4]. One of the components in analytical thinking is a viable mental model. This is important because a computer programmer needs to form a viable mental model of the problem and how to solve the problem before attempting to write a program [5].

RELATED LITERATURE

Adelson and Soloway [6] reported that 30 to 50 percent of the Computer Science students from universities around the world have failed in computer programming at a basic level. McCracken [7] found that first year students only scored an average of 22.89 marks out of 110 marks in a programming test. Poor performance in programming is the most contributors to the failure in Computer Science course [8].

Difficulties in Learning Programming

A study conducted by Winslow [9] found that students may be able to differentiate between syntax and semantics in different statements, but they do not know how to combine both of them into a meaningful program. This study was supported by Ala-Mutka [10] when she found that students can learn to understand and explain a programming concept verbally, but they failed to apply the concept in writing program. Even if they can solve a problem in writing, but they still failed to translate the solutions into a program. Whereas, in different studies by Tan et al [11] have shown that undergraduate students who took subject of Computer Programming 1 at Multimedia University of Malaysia

Corresponding Author: Hawariyah Rahim, 1Jabatan Teknologi Maklumat dan Komunikasi, Politeknik Tuanku Syed Sirajuddin, Perlis, Malaysia hawariyah@gmail.com

have problems in designing a program to solve a given problem. Studies conducted by Matthiasdottir and Geirsson [12] also found that students can understand program code written by others but they have difficulties in translating their own solutions to meaningful program.

Problem-Solving Skills

Mendes [13] found that students encounter problems in applying the concepts of programming to solve problems especially in engineering area. Lister et.al [14] also found that students are lack of problem solving skill. In different studies, Sleeman [15] found that many students in Introduction to Programming class not know how to code a program to select and display the smallest number in a list of integers. There are two-phase process in writing a program; problem-solving phase and implementation phase. Three main steps in problem-solving phase consist of analysis and specification, general solutions (algorithm) and verify [1][2]. The second phase consists of concrete solution (program) and test. It shows that writing an algorithm is an important step for students to solve a problem in programming. In addition, a student needs to think analytically to write a good algorithm and solve a programming problem [3][4].

Analytical Thinking

Analytical thinking is a critical component in visual thinking that provides the ability to solve problems in a quick and effective way. It is a step-by-step thinking that enables a student to solve complex problems into a single and controlled component. Analytical thinking is closely related to visual thinking. It is a part in problem-solving process. Whereas, visual thinking is related to mental model where it helps to explain the model itself more clearly and distinctly [16].

Mental Model

Craik [17] defines mental models as models in small scale formed by human mind to anticipate events, to make reasoning and to become the basis of the description of an item. Laird et.al [18] also defines mental models as psychological representation of the real situation, based on assumptions or imaginary. Borgman [19] said that mental model is a cognitive mechanism built by human to describe and draw conclusions about a system or problem and give reaction in the real world. Generally, mental model is an internal representation (mental) on a real system including behavioral, organizational and internal structure of the system.

Mental Models in Programming

Balzert [20] and Rist [21] argues that programming requires a person to store various types of information in his memory as detailed information about the syntax and semantics of a programming language, some mental model for problem solving and the ability to distinguish between a specification to solution and a solution to a problem. His opinion was supported by Lui et.al [22] who said that programming involves the construction of some mental models.

INITIAL STUDY

A study was conducted by interviewing six experts in the field of programming language subject consists of two university lecturers, three polytechnic lecturers and a high school teacher who have been teaching programming in more than five years. Five of the six experts are actively involved in research related to the field of programming. The main objective of the interviews conducted in this study is to identify major problems in teaching and learning programming. The data was collected using structured interview method which consist a list of three predetermined questions. The interviews were conducted face to face with the interviewees and over the telephone. The first question was to find out the difficulties in learning programming language among undergraduate students. The second question was to find out what are the current methods or techniques used to teach programming language in local universities and polytechnics. And the third question was to find out what are the suitable methods or techniques that can be used to solve the problems in learning programming. The data gathered from the interviews are then analyzed using domain or taxonomy analysis method. All the answers obtained from the interviews are transformed and coded into a transcript to build themes and sub-themes. Then, the themes are organized to make meaning connections regarding the objective.

FINDINGS AND DISCUSSION

The results from the analysis are shown in Table 2 and Figure 1. There are eight different categories of problems that are arranged in descending order. From the graph, problem-solving skill is the most frequently mentioned in the interviews followed by problem that involved understanding the basic concepts of programming. The rest are the problems in the syntax

and semantics of programming, basic math skills, individual learning strategies, students' ethics, traditional teaching and paradigm change. Further information of the problems is described in the next section.

Table 2: Frequency of different types of difficulties in learning programming

Types of difficulties	Frequency
Problem-solving skills	10
Understanding basic concepts of programming	5
The syntax and semantics of programming	3
Basic skills in mathematics	3
Individual learning strategies	2
Students' ethics	2
Traditional teaching	1
Change of paradigm	1

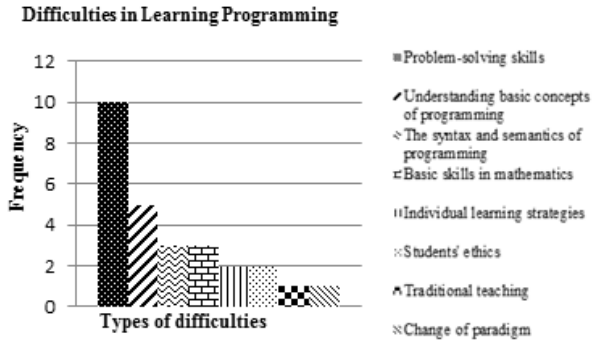


Figure 1: Types of difficulties in learning programming

From the interviews, interviewees elaborated the problems faced by the students in problem-solving skill as follows:

a. Students are difficult to translate the answer into a programming language.
b. Students have difficulties in writing a program.
c. Students are unable to answer the question in the form of program.
d. Students do not know how to translate data into program.
e. Students do not know how to translate questions into program.
f. Students cannot imagine the data in the form of program.
g. Students do not know how to apply their programming knowledge into program.
h. Students are difficult to interpret natural language into the machine language.
i. Students lack of critical thinking skills.
j. Students memorize a lot rather than thinking.

Students have problems the following problems while learning to understand basic concept of programming:

a. Students are confused to declare data.
b. Students do not have a solid foundation in programming.
c. Students do not apply the step by step process in programming.
d. Students did not know how to read and understand the code of the program.
e. Students do not understand the meaning of code lines in program.

To understand the syntax and semantic of programming is also a big challenge or for students. Students do not understand the syntax and semantics, students have difficulties in syntax and semantics and students have difficulties in manipulating the code of the program are the input given by the respondents. Students that weak in logic operations, students that have problem in logic and students that weak in mathematics are the problem associated with strong skills in mathematics.

Individual learning strategies, students' ethics, traditional teaching and change of paradigm are the problems with the lower frequencies. Lacks of practicing after classroom and students cannot solve problems in form of 'tracing' are categorized as problems in individual learning strategies. Problems also occurred when students have to change their thinking from structured programming concept into object oriented programming concept.

The interviews also reveal the methods of teaching programming such as slides, programming software, notes and sample questions. These traditional teaching methods are frequently used in programming classes. Moreover, support system simulation and computer game are recommended to improve the existing teaching methods.

CONCLUSION

From the interviews, it shows that students are having difficulties to translate solution into program. This problem is closely related on how a student builds a viable mental model to form a correct algorithm to write a program and to solve a programming problem. Writing a good algorithm is an important skill which contained in problem-solving phase. To write a good algorithm

and to solve a problem, a student needs to think analytically. Constructivism approach can help students to build a good and stable mental model. However, learning through constructivism approach alone is not enough to improve mental model of a student. Visualization programs such as game are suitable as a learning material because it can encourage students to learn collaboratively in an entertaining environment and increase their motivation to learn programming language. More research needs to be conducted to determine the use of constructivist approach and games to improve students' mental model in programming.

REFERENCES

[1] Blank, G & Barnes, R.1998. The Universal Machine.MA: WCB/McGraw Hill.Chapters 2.1.Boston

[2] Dale, N., McMillan, M., Weems, C., & Headington, M.R., 2003. Programming And Problem Solving With Visual Basic.NET. Jones and Bartlett Publishers. Boston

[3] Estivill-Castro, V., & Bartlett, B. J. 2011. Image Capture for Concrete Programming: building schemata for problem solving. In CSEDU 2011. http://www. csedu.org/CSEDU2011/.

[4] Munir, 2001. Pembangunan dan Keberkesanan pakej Multimedia dalam Pendidikan untuk memotivasikan Literasi (MEL) di kalangan kanak-kanak prasekolah. PHD Thesis,Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia.

[5] Robins, A., Rountree, J., & Rountree, N. 2003. Learning and Teaching Programming: A Review and Discussion. Computer Science Education, 13(2), 137–172.

[6] Adelson, B., & Soloway, E. 1985. The Role of Domain Experience in Software Design. Software Engineering, IEEE Transactions on, (11), 1351-1360.

[7] McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y. B. D., & Wilusz, T. 2001. A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. ACM SIGCSE Bulletin, 33(4), 125-180.

[8] Denning, P. J., & McGettrick, A. 2005. Recentering computer science. Communications of the ACM, 48(11), 15-19.

[9] Winslow, L. E.1996. Programming pedagogy—a psychological overview. ACM SIGCSE Bulletin, 28(3), 17-22.

[10] Ala-mutka, K. 2004. Problems in Learning And Teaching Programming - A Literature Study For Developing Visualizations In The Codewitz-Minerva Project. Codewitz Needs Analysis, 1–13.

[11] Tan, P.-H., Ting, C.-Y., & Ling, S.-W. 2009. Learning Difficulties in Programming Courses: Undergraduates' Perspective and Perception. International Conference on Computer Technology and Development (pp. 42–46). Ieee. doi:10.1109/ICCTD.2009.188

[12] Matthiasdottir, A., & Geirsson, H. J. 2011. The Novice Problem in Computer Science. CompSysTech'11: Proceedings of the 12th International Conference on Computer Systems and Technologies (pp. 570–576).

[13] Mendes, A., Vilner, T., & Nordström, M. 2005. Difficulties Teaching Java in CS1 and How We Aim to Solve Them. ITiCSE'05: Proceedings of the 10th annual SIGCSE conference on Innovation and technology in computer science education (pp. 344–345).

[14] Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M. & Thomas, L. 2004. A multi-national study of reading and tracing skills in novice programmers. ACM SIGCSE Bulletin, 36(4), 119-150.

[15] Sleeman, D. 1986. THE CHALLENGES OF TEACHING COMPUTER. Communications of the ACM, 29(9), 840–841.

[16] Rist, R., & Terwilliger, R. 1995. Object-oriented programming in Eiffel. Englewood Cliffs, NJ: Prentice Hall.

[17] Craik, K. J. W. 1967. The nature of explanation. CUP Archive.

[18] Johnson-Laird, P. N. 1983. Mental models: Towards a cognitive science of language, inference, and consciousness (Vol. 6). Harvard University Press.

[19] Classroom: A Theoretical Model of Cognitive Conflict Process.

[20] Borgman, C. L. 1986. The user's mental model of an information retrieval system: an experiment on a prototype online catalog. International Journal of man-machine studies, 24(1), 47-64.

[21] Balzert, H., Balzert, H., & Zwintzscher, O. 2004. Die E-Learning-Plattform W3L. Wirtschaftsinformatik, 46(2), 129-138.

[22] Lui, A. K., Kwan, R., Poon, M., & Cheung, Y. H, 2004, Saving Weak Programming Students: Applying Constructivism In A First Programming Course. SIGCSE Bull. 36(2), pp.72-76.