



## Effect of Hybrid Teaching Methodology and Student Group Policy on Object Oriented Problem Solving

Kanar Shukr Mohammed<sup>1</sup>, Moayad Y. Potrus<sup>2</sup>, Bnar Faisal A. Daham<sup>3</sup>

Department of Software Engineering, College of Engineering, Salahaddin University, Erbil, Kurdistan Region, Iraq.

### ARTICLE INFO

#### Article History:

Received: 02/10/2017

Accepted: 10/09/2018

Published: 28/10/2018

#### Keywords:

Object Oriented

Programming,

Teaching Methodology

Software Group Policy

#### \*Corresponding Author:

Kanar Shukr Mohammed

kanar.muhamad@su.edu.krd

### ABSTRACT

Teaching programming at university for beginner's level represents one of the most difficult tasks for faculty members. It has many challenges such as wide diversity of student's background knowledge, not able to understand programming concept, identifying programming language syntax usage etc. Through the literature, many methods and tools have been covered to make programming concepts easier to understand for students. In this paper, hybrid approach is proposed to efficiently increase the understanding capacity of the students. The hybrid method is based on problem-based and puzzle-based learning (PBL & PZBL) methods. In order to test the proposed approach, students enrolled in object oriented programming courses in software engineering department were considered as test case. In addition, the paper addresses the benefits of using non-fixed student group policy with the proposed hybrid method to get a better understanding of student learning outcome. The proposed method scored a better result when compared to standalone methods. The enhancement was reflected both in student questionnaire and final course work results.

### 1. INTRODUCTION

Programming languages represents one of the most important topics in computer science. It is a core subject where students will gain the necessary tools for creating software programs. In software engineering, programming language represents the core tool used during development phase of software development cycle. Therefore, it is very crucial for developers to have a very good knowledge in programming language, especially, in object oriented programming, as it is the main programming paradigm used in developing windows, web and mobile based applications (Eckerdal, 2006).

Learning object oriented programming (OOP) is one of the core requirements for every modern developer. It is one of the main subjects in computer and software related departments where learners or students will gain the required skills for understanding the concepts of OOP. However, teaching this subject is one of the main challenges for academic staff, as most of the students will find it difficult to understand (Pereira, 2010). Therefore, many researches in the field of teaching OOP emerged to establish the best tool or methodology to simplify the subject, as well as, to motivate the student and get them

involved in the learning process (Teague, 2007).

In this paper, a hybrid algorithm based on two well-known strategies which are Problem-based and Puzzle-based learning methodologies. The proposed strategy tries to merge the strength of both strategies so that a maximum learning outcome is achieved. The remaining of this paper is organized as follows, in section 2, a comprehensive literature is presented that covers the tools and methods used in teaching programming as general and OOP as special. In section 3, the proposed hybrid method based on Problem-based Learning (PBL) and Puzzle-based Learning (PZBL) is introduced. Section 4, covers the result and discussion of applying the proposed strategy. Finally, the final driven conclusion is covered in section 5.

## 2. RELATED WORKS

Teaching methodology in programming was initiated at the early 80's. The subject importance was gaining momentum because of the use of computers in various aspect of life (Nuutila, 2008). The focus was on procedural and low-level programming at college level to help students gain maximum knowledge on the subject (Montero, 2010). However, during 90's, when OOP based languages, such as Java, C++ and ADA, started to spread rapidly, researchers started to find new methods and tools to teach programming (Pilkington, 2010; Looi, 2014; Lykke, 2014).

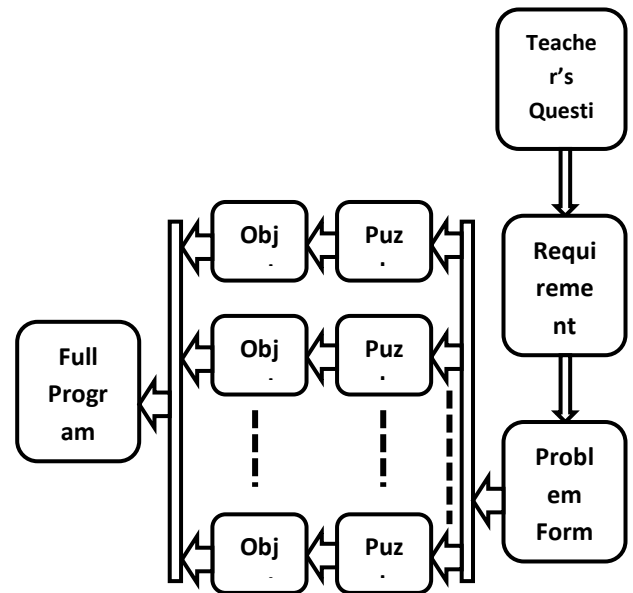
There are numerous papers and researches on teaching methodology in programming. The main focus of these papers is either to introduce the students with new teaching paradigms, or designing a special tool for simplifying the understanding of the subject (Vogel-Heuser, 2012; Teague, 2007). OOP languages such as Java and C++, has been used to develop various applications and software

for many years. The methodologies used with OOP paradigm varied from problem based, buzzle based, game based and hybrid based (Ala-Mutka, 2005; Bellstrom, 2009; Fee, 2010; Montero, 2010; Dietrich, 2014). The problem based methodology was basically used to motivate medical school students to find a solution for any problem they face during their medical education (Beaumont, 2003; Dijkstra, 1989). Each problem must pass through seven steps to find a proper solution. These steps are 1) terms and concept clarification; 2) problem identification; 3) brainstorming; 4) explanatory model sketching; 5) learning issue formulation; 6) self-learning and 7) information synthesising and testing (Falkner, 2010). These concepts can be easily integrated in programming problems, as the students will try to solve any problem by writing a program (Kay, 2000). These researches have been motivated by the increased importance of programming languages in computer science and software engineering (Nuutila, 2007; Falkner, 2010, Livovsky, 2014). In PBL for Programming, the problem is also seen as a learning core for student to understand how problem is formed, analyzed and solved (Nuutila, 2007; Malik, 2017). The process involves a learning scenario (Yoneyama, 2008), or concerning a large scale problem (Livovsky, 2014; Montero, 2010) or related to learning outcomes of the course (Nuutila, 2007; Livovsky, 2014; Kramer, 2016; Thota, 2010; Sooriamurthi, 2015). The algorithm also was integrated fo some challenging problems (Livovsky, 2014; Kramer, 2016, Malik, 2017), or complex problem (Peng, 2010), and ill-structured. The features of the studied solutions may relate to special type of problems that requires an open-mindedness (Mathews, 2013; Santos, 2013), or realworld simulation (Kramer, 2016; Peng, 2010), activation of prior knowledge (Thota, 2010; Santos, 2013), integration of theory and practice (Thota, 2010), increase of problem

complexity and sufficient solution for a duration of time (Karmer, 2016), followed by validation (Malik, 2017). However, the study found several problematic characteristics in PBL. Due to time and erratic changing of the curriculum constraint in Computing courses, problems that are well-defined, well-structured, and uncomplicated are much more preferable (Yoneyama, 2008).

Puzzle based learning (PZBL) is a creative way of learning in which students deals with any programming problem as a puzzle (Pereira, 2010). The idea is to initiate the thinking of students about a problem in a way that it will motivated them to think about it in different way (Redmond, 2001). There was a various researches in the field of puzzle based problem for various problems, however, there were few that dealt with programming in computer science and engineering courses (Pereira, 2010), or by using it for OOP language learning (Merrick, 2010), or to evaluate its impact on introductory courses in computer science (Redmond, 2001). It was adapted with student learning history in programming to efficiently enhance the outcome of student programming learning process (Tuikys, 2016). It was also used to help students understand the design concept of Design Patterns for software engineering students (Rusu, 2011).

**Figure 1: Proposed system for Hybrid learning method for OOP programming**



In this research, a hybrid learning method is proposed to solve various programming problems in OOP. The method uses both methods to consider some of the problems studied in second year software engineering. The following sections will thoroughly explain the methodologies used in this research.

**3. PROPOSED METHOD**

The following Fig. 1, explains the steps used in this research. The steps show how the question forms a problem and then the problem

is divided into puzzles and finally is merged to form one program.

In the first step, the question is initiated by the teacher for the students to find a programming solution. Then, the students will try to find a problem-based form of the question such that the requirements are clear and the final targets are specified. These specified requirements are a little hard to understand from programming view and need to be simplified. In most cases, the problem in hand can be complex where multiple independent requirements are formed. Therefore, each problem is divided into smaller sub problems which can be easily understood, and clearer set of requirements are obtained for each one.

The puzzle-based learning is used to transform these requirements from a problem-based to a puzzle per sub-problem. These puzzles are easier to understand by the students and will form a brain challenge for them to find a proper solution (O'Grady, 2012). Hence, an object (class) will represent each puzzle. These objects are programmed and tested independently and therefore, can be assigned to different groups to solve each one of them. Finally, all these puzzles (objects) are merged

together to form one big program which then is tested for integration and verified against meeting the initial requirements.

## **4. RESULTS AND DISCUSSION**

### **4.1 Experiment setup**

The proposed system was integrated in software engineering second and third year courses of Object Oriented Programming and Mobile Application using Android during the academic years between 2013-2015. Both courses use Java as a programming language which is an OOP programming based language. The students were setup in groups that can form sub groups within it. During the subject curriculum, the students were presented with various type of software systems which varies in difficulty, starting with simple problems and finishing with complex systems. In the first study year the problem-based learning was integrated in the studying system without the puzzle-based learning. While in the second year only puzzle-based learning was integrated. Finally, the proposed method was integrated in the third year.

At the end of the courses a questionnaire was distributed among the students to find the impact of the learning methods on the students and the feedback was used for further assessment. Table 1 lists the questionnaire used and the question types asked during the process. It is worth noting, that the student groups were selected randomly and fixed during the experiment to guarantee the correctness of the results.

### **4.2 Results and discussion**

The result of the questionnaire is illustrated in Table 1 for the above questions with the results clearly demonstrated that the proposed method has a good impact on student learning outcome.

**Teaching Method Evaluation and Improvement**

Q1. The aims of the teaching methods were clear to me  
 a. Very Clear  
 b. 50 % Clear  
 c. Not Clear

Q2. The teaching methods was challenging and interesting  
 a. Very much  
 b. Not that much  
 c. Not interesting at all

Q3. The teaching methods provided effective opportunities for active student participation in learning activities  
 a. Very Good  
 b. Good  
 c. Fair  
 d. Not Good

Q4. The teaching methods was effective for developing my thinking skills (e.g. critical analysis, problem solving)  
 a. Very Good  
 b. Good  
 c. Fair  
 d. Not Good

Q5. The information/materials provided for the teaching methods helped me to learn independently.  
 a. Very helpful  
 b. 50% helpful  
 c. Not helpful

Q6. The aims of the teaching methods were met  
 a. 100%  
 b. 50%  
 c. Not met

Q7. Overall, I was satisfied with the quality of the teaching methods  
 a. Very much  
 b. Not that much  
 c. Not satisfied at all

Q8. Which of them are the best methods used for learning:  
 a. Problem-based  
 b. Puzzle-based  
 c. Puzzle & Problem Based

Q9. The best features of the teaching methods were...

Q10. The teaching methods could be improved by...

**Figure 2: The questionnaire used for student feedback about the proposed hybrid method compared to the problem-based and puzzle-based learning**

Out of 32 students, 19 students preferred the proposed method over the problem-based (6 students) and puzzle-based (5 students). In addition, the method feedback showed a better impact in the sections of learning understanding, develop thinking skills, clearness and satisfaction.

In addition to the above questionnaire, the impact of the proposed method where observed in the field of student improvement during exams. The students were divided into 7 groups with each group consists of 4-5 students and these students can be sub-grouped into 2-3 subgroups.

**Table1: Questions and Result of the Questionnaire about the Proposed Method**

Question	a	b	c	d
----------	---	---	---	---

The aims of the teaching methods were clear to me?	16	15	1	
The teaching methods were challenging and interesting	16	13	3	
The teaching methods provided effective opportunities for active student participation in learning activities	5	22	4	1
The teaching methods were effective for developing my thinking skills (e.g. critical analysis, problem solving)	7	18	6	1
The information/materials provided for the teaching methods helped me to learn independently	20	12	0	
. The aims of the teaching methods were met	4	27	1	
Overall, I was satisfied with the quality of the teaching methods	11	19	2	
Which of them are the best methods used for learning?	6	5	19	2 no vote

The practical testes where introduced to the groups as a part of their activities during lab sessions, such that, each week, the student attempt to solve the same problem using problem-based, puzzle-based and hybrid method. Then, the time and mark was registered (out of 10) for each method to find the improvement in student performance. The following Tables 2 and 3 show the seven test results of problem-based and proposed hybrid method for seven group of students. The tests various from simple to complex problem.

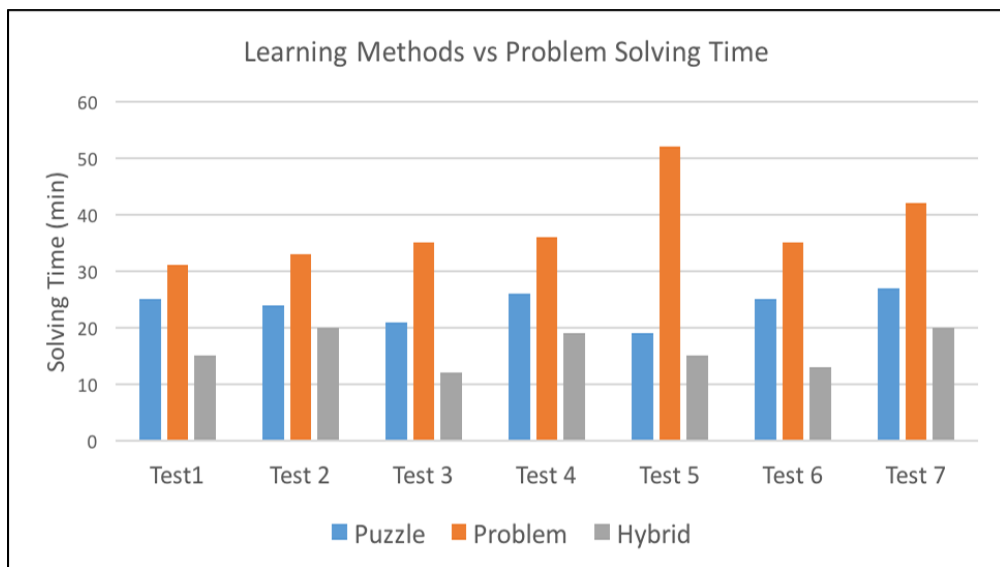
**Table 2: Problem-Base Learning Test Results**

Group	Test1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Average Mark
1	8.3	8.6	7.1	9.3	10.0	10.0	10.0	7.87
2	6.0	7.4	8.1	10.0	10.0	9.7	2.8	8.76
3	7.7	7.7	3.9	8.3	10.0	7.2	10.0	9.44
4	8.0	8.6	7.4	10.0	8.9	7.2	6.4	9.61
5	9.7	8.9	9.0	9.7	10.0	7.8	10.0	7.14
6	10.0	10.0	7.1	10.0	10.0	9.7	6.4	8.53
7	10.0	8.0	7.4	10.0	7.2	9.7	9.5	8.46

**Table 3: Hybrid Learning Test Results**

Group	Test1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7	Average Mark
1	9.3	8.6	9.4	10	10.0	10.0	9.7	9.19
2	10.0	9.4	9.0	10	10.0	10.0	10.0	9.96
3	9.7	8.9	9.7	10	10.0	10.0	10.0	9.37
4	10.0	9.7	9.4	10	10.0	10.0	10.0	10.00
5	10.0	9.7	9.0	10	8.9	10.0	10.0	9.29
6	8.3	9.7	9.7	10	7.8	10.0	10.0	9.43
7	7.3	10.0	9.4	10	8.3	9.7	4.6	9.23

From the above two tables, it is obvious that the average mark and group marks shows a great improving in marks except for one of the tests. The improvement reflects the easier understanding of the problem when it is divided into multiple puzzles to form objects. This is also reflected in total time required to solve a problem using the proposed method



**Figure 3: Learning Method vs Problem Solving Time (ms).**

compared to problem-based learning method as illustrated in table 4 and Fig. 3.

Table 4: Comparing Solving Time (min) for each Learning Method

Group	Test1	Test 2	Test 3	Test 4	Test 5	Test 6	Test 7
Puzzle	25	24	21	26	19	25	27
Problem	31	33	35	36	52	35	42
Hybrid	15	20	12	19	15	13	20

The main reason beyond the long time in problem-based learning compared to the other two methods, is that, the problem based method gives the problem as an overall requirement which the student needs to understand and break to pieces. This process is Thus, simplifying the problem in this form makes it understandable and easier to understand the purpose of OOP programming. Finally, it is worth mentioning that not all problems will prove solvable and understandable using the proposed method. In some cases, when the problem is much simpler, the student can solve the problem using one object, and thus the method loses its efficiency.

## 5. CONCLUSION

Teaching programming to university students represent one of the great challenges facing lecturers in computer related department especially OOP. This paper presents a hybrid method using both problem-based learning and puzzle-based learning methods. The method works in forming a basic problem with

the most important and difficult part of programming. Hence, the student will take much time to analyze the problem. Meanwhile, in puzzle-based learning, the student will deal with a set of requirements as a puzzle, which make it much easier to understand than problem-based but it will take a little longer for student to form the problem statement. Finally, the hybrid method will make it simpler to the student to obtain the problem, divided into pieces of puzzles and creating an object for each piece of them. Therefore, the total time will be much shorter than the other two methods.

In general, it has been shown through this paper that, the proposed hybrid method proved to be more efficient than both problem-based and puzzle based used by many researches to teach programming language especially OOP. The method strength comes from the strategy used in forming the main problem into a set of puzzles and making each puzzle works as an object.

requirements from the question in hand. Then, this problem is broken into a set of puzzles and these puzzles will form an individual object. This object will be easier to program and test compared to the basic method. The method was tested on a group of student studying OOP in software engineering and the obtained results showed the improvement in student learning level compared to stand alone problem and puzzle based methods.

## REFERENCES

- Ala-Mutka, K. 2005. 'A Survey of Automated Assessment Approaches for Programming Assignments'. *Computer Science Education* 15:2, pages 83-102.
- Beaumont, C and Frank, B. 2003. 'Enhancing Employability through PBL. Delivering Employability'. *The Diversity of Approaches in*

- Higher Education Conference, University of Central Lancashire, Preston,.
- Bellström, P; Kilbrink, N. 2009. 'Problem-Based Learning in a Programming Context—Planning and Executing a Pilot Survey on Database Access in a Programming Language'. In *Information Systems Development*, pp. 867-875.
- Dietrich, S. 2014. 'A metaphor for understanding objects and classes in CS1'. *ACM Inroads*5:4, pages 42-43.
- Dijkstra, E. W. 1989. 'On the Cruelty of Really Teaching Computing Science'. *Comm. ACM* 32: 1398-1404.
- Eckerdal, A; McCartney, R; Moström, J; Ratcliffe, M; Zander, C. 2006, 'Categorizing student software designs: Methods, results, and implications'. *Computer Science Education* 16:3, pages 197-209.
- Falkner, N; Sooriamurthi, R; Michalewicz Z. 2010. 'Puzzle-Based Learning for Engineering and Computer Science', *Journal Computer*, Vol. 43, Issue 4, USA, pp. 20-28.
- Fee, S and Holland-Minkley, A. 2010. 'Teaching computer science through problems, not solutions'. *Computer Science Education*, vol.20, pp. 129–144.
- Kay, J; Barg, M; Fekete, A; Greening, T; Hollands, O; Kingston, J; Crawford, K. 2000. 'Problem-Based Learning for Foundation Computer Science Courses'. *Computer Science Education* 10(2): pages 109-128.
- Kramer, M; Hubwieser, P; Brinda, T. 2016. 'A Competency Structure Model of Object-Oriented Programming'. 2016 International Conference on Learning and Teaching in Computing and Engineering (LaTICE), pages 1-8.
- Livovský, J and Porubän, J. 2014. 'Learning object-oriented paradigm by playing computer games: concepts first approach'. *Open Computer Science* 4:3.
- Looi, H J and Seyal, A. 2014. 'Problem-based Learning: An Analysis of its Application to the Teaching of Programming'. *International Proceedings of Economics Development and Research*, vol. 70, p. 68–75.
- Lykke, M; Coto, M; Mora, S; Vandel, N and Jantzen, C. 2014. 'Motivating programming students by problem based learning and LEGO robots', 2014 IEEE Global Engineering Education Conference (EDUCON), Istanbul, 2014, pp. 544-555.
- Malik, S and Coldwell-Neilson, J. 2017. 'A model for teaching an introductory programming course using ADRI'. *Education and Information Technologies* 22:3, pages 1089-1120.
- Malik, S and Coldwell-Neilson, J. 2017. 'Impact of a New Teaching and Learning Approach in an Introductory Programming Course'. *Journal of Educational Computing Research* 55:6, pages 789-819.
- Matthews, R; Hin, H S; Choo, K A. 2013. 'Novice Programming Students' Perception of Learning Object'. 2013 International Conference on Informatics and Creative Multimedia, pages 292-297.
- Merrick, K E. 2010. 'An Empirical Evaluation of Puzzle-Based Learning as an Interest Approach for Teaching Introductory Computer Science', *IEEE Transactions on Education*, Vol. 53, No. 4, pp. 677-680.
- Montero, S; Diaz, P; Diez, D; Aedo, I. 2010. 'Dual instructional support materials for introductory object-oriented programming: Classes vs. objects'. *IEEE EDUCON 2010 Conference*, pages 1929-1934.
- Nuutila, E; Törmä, S; Kinnunen, P; Malmi, L. 2008. 'Learning Programming with the PBL Method - Experiences on PBL Cases and Tutoring'. In J. Bennedsen (Ed.), *Issues in Introductory Programming Courses* (pp. 47–67). Berlin Heidelberg: Springer-Verlag.
- Nuutila, E; Törmä, S; Malmi, L. 2007. 'PBL and Computer Programming — The Seven Steps Method with Adaptations', *Computer Science Education*, 15:2, 123-142.
- O'Grady, M. 2012. "Practical Problem-Based Learning in Computing Education". *ACM Transactions on Computing Education (TOCE)*, vol.12, pp. 10.
- Peng, W. 2010. 'Practice and experience in the application of problem-based learning in computer programming course'. *Educational and Information Technology (ICEIT)*, 2010 International Conference on. Vol. 1. IEEE.
- Pereira, H; Zebende, G and Moret, M. 2010. 'Learning computer programming: Implementing a fractal in a Turing Machine'. *Computers & Education*, vol.55, pp. 767-776.
- Pilkington, C and Gelderblom, H. 2010. 'Using the Karplus learning cycle to teach learners with ADHD introductory computer programming'. *African Journal*

- of Research in Mathematics, Science and Technology Education 14:1, pages 73-84.
- Redmond, M. 2001. 'A computer program to aid assignment of student project groups'. In Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education (SIGCSE '01). ACM, New York, NY, USA, 134-138.
- Rusu, A; Russell, R; Cocco, R; Di Nicolantonio, S. 2011.' Introducing object oriented design patterns through a puzzle-based serious computer game', 2011 Frontiers in Education Conference (FIE), Rapid City, SD, pp. 1-6.
- Santos, S and Garcia, V C. 2013. 'PBL in Teaching Computing: An Overview of the last 15 years'. Frontiers in Education Conference, p. 123 – 142.
- Sooriamurthi, R; Falkner, N; Meyer, E and Michalewicz, Z. 2015. 'Puzzle-Based Learning: Introducing Creative Thinking and Problem Solving for Computer Science and Engineering'. In Proceedings of the 46th ACM Technical Symposium on Computer Science Education (SIGCSE '15).
- Teague, D and Roe, P. 2007. 'Learning to Program: Going Pair-Shaped'. Innovation in Teaching and Learning in Information and Computer Sciences 6:4, pages 4-22.
- Thota, N; Whitfield, R. 2010. 'Holistic approach to learning and teaching introductory object-oriented programming'. Computer Science Education 20:2, pages 103-127.
- Tuikys, V; Burbait, R; Bepalova, K; Ziberkas, G. 2016. 'Model-driven processes and tools to design robot-based generative learning objects for computer science education'. Science of Computer Programming 129, pages 48-71.
- Vogel-Heuser, B; Braun, S; Obermeier, M; Jobst, F; Schweizer, K. 2012. 'Usability evaluation on teaching and applying model-driven object oriented approaches for PLC software'. 2012 American Control Conference (ACC), pages 4463-4469.
- Yoneyama, Y; Matsushita, K; Mackin, K J; Ohshiro, M; Yamasaki, K; Nunohiro, E. 2008. 'Puzzle Based Programming Learning Support System with Learning History Management', Proceedings of the 16th International Conference on Computers in Education, pp. 623-627.