Problem-based Collaborative Learning Strategy in Computer Programming

Irwan Irwan ¹, Wilda Susanti ², Yenny Desnelita ¹, Gustientiedina Gustientiedina ², Fery Wongso ³, Ahmad Fudholi ⁴,⁵

¹ Prodi Information Systems, Institut Bisnis dan Teknologi Pelita, Indonesia
² Prodi Technical Information, Institut Bisnis dan Teknologi Pelita, Indonesia
³ Prodi Information Systems STMIK Dharmapala, Indonesia
⁴ Universiti Kebangsaan Malaysia 43600 UKM Bangi, Selangor, Malaysia
⁵ National Research and Innovation Agency Republic of Indonesia (BRIN), Bandung, Indonesia

Abstract – Computer programming is the basic skill that has to be possessed by students who attend Algorithm and Programming courses. This skill requires a problem solving strategy and involves a large number of programming logic activities. The purpose of this study is to offer a collaborative learning environment in order to provide a learning experience by interacting socially. However, social interaction does not just happen; the appropriate guiding strategies are needed to support collaborative learning. The proposed strategy is a problem-based practice strategy. This research was conducted on the second semester students of the Informatics Engineering Study Program, Faculty of Computer Science, Pelita Indonesia. Two classes were used for testing, namely the control class with 16 students and the experimental class with 15 students. The control class is taught by conventional collaborative learning and the experimental class is taught by collaborative learning using problem-based practice strategies.

The results show that the proposed strategy can be useful for students in improving computer programming skills in a collaborative learning environment with a problem-based approach.

Keywords – Practice strategy, collaborative learning, learning experience.

1. Introduction

Basic programming is a must-have skill for students who want to learn the program. Learning computer programming requires students' motivation and involvement in the learning process by building more effective mechanisms and tools for programming skills [1]. Studying computer programming in a team is closely related to creative thinking, summarizing systematically, and communicating and collaborating with peers [2]. Learning programming requires students to improve creativity, team work, innovation and knowledge of data structures and algorithms [3]. Computer programming skills require several types of thinking skills such as logical thinking and problem solving [4]. Developing a computer program requires the ability to translate and model a person's way of thinking, problems and solutions in natural language into the chosen programming language [5]. Computer programming requires problem-solving strategies and involves many logical programming activities that pose challenges for students [6]. Programming is the process of writing, testing and debugging, and maintaining the code that builds a computer program. To do programming, skills in algorithms, logic and programming languages are needed [7].

In fact, for students who do not have knowledge of computers, learning program is something that is difficult [8]. It is a challenge how to provide learning strategies to improve the cognitive abilities of students in learning computer programming [9]. From the literature review, researchers have done a lot of research to improve beginners' understanding of computer programming [10], [11]. The high cognitive load that students have, so they have to have more skills and continue to practice over and
over again. This leads to frustration and loss of interest and enthusiasm for programming [12], [13].

Group learning using technology is the current preferred learning approach. This approach makes it easier for students to do difficult tasks with the help of social interactions [14]. Collaborative learning increases group knowledge, improves problem solving skills and motivates students to participate in learning [15], [16]. Expertise in collaborative computer programming has been demonstrated by several researchers [17]. On the other hand, educators also find that, on many occasions, learning in teams does not bring the expected social interactions. In addition, students' achievement may be disappointing due to a lack of learning strategies [18]. Therefore, how to improve student social interaction in a collaborative learning environment has become an important issue.

Collaborative learning is a learning method in which small groups whose members have the same position collaborate to achieve common goals, to perform common tasks and to evaluate joint results [19]. The characteristics of collaborative learning in the classroom [20] are: (1) communication is the way students share, exchange, and grow knowledge; (2) social interdependence, namely the dependence of students on each other to find important knowledge and participation for each student; (3) student exploration which is a student's contribution to discovering knowledge that does not only rely on lecturer presentations; (4) promote the interaction that can be done by starting the activity with the problem; (5) diversity between groups means that each group consists of different levels of learning ability; (6) an assessment that is not only based on group performance but also individual performance.

In research [6] the practice of learning computer programming was divided into two groups to work in teams with problem-based practice strategies. The lecturer gives program pieces to be completed by team members and one program is proposed by other team members. During the learning activities there is feedback from the lecturer. Students review the code provided by the lecturer and correct their own program [21]. After that, the lecturer answered the questions asked by the students. Interviews from selected universities have been conducted for a collaborative learning approach. The purpose of this study was to determine the significant correlation of pretest and posttest scores of students in increasing computer programming skills. In this case, the researcher tries to explain collaborative learning by using problem-based methods to students of the Informatics Engineering Study Program as an effort to improve student abilities in the Algorithm and Programming course material.

2. Method

This research belongs to experimental research design. The research design used is a classical experimental design. The population in this study were 31 students of the Informatics Engineering Study Program who took the Algorithm and Programming course for the 2020/2021 academic year. Students were divided into two groups, the experimental group comprised 15 people, and 16 people were included into control group. This study uses the C++ programming language.

The type of research model carried out refers to the model developed by Wang and Hwang [6]. The description of the steps of the learning model which was carried out can be seen in Figure 1.

Integration of the Wiki module is done in the Moodle LMS as part of learning management. LMS is a popular learning platform today [22]. Learning activities in this environment is performed that each student in the team can edit, provide comments and can record each edited result.

Grouping is done by lecturers to give assignments and to be able to exchange assignments between teams. There are five steps in the activities within learning activities. In the first step, each team member enters the learning content that has been provided. The assignment was carried out by the lecturer to two teams. The team did a coding analysis. Two Wiki pages are owned by each Team in Moodle, one for issue sub-missions and the other for code analysis. An example is given on the problem presentation page. Each team works according to the direction of the lecturer where the wiki page for program analysis can be entered by other teams.

Second step comprises asking a problem after students understand and can complete programming assignments by relating it to their experience. Interface for the system is done by using the C++ programming language.

In the fourth step is to do the programming themselves and collaborate for the final result. During the fifth step, the lecturer provides feedback about programming, provides solutions and program code. Team members adapt the program according to suggestions given by other groups. Each group completes the problem-posing stage after which it enters the code-submission stage, two groups work on each other's tasks and give each other feedback.
3. Results and Discussion

Student learning activities are guided by lecturers in Algorithm and Programming courses including C++ syntax. An initial test in the form of a questionnaire was given to students to see their understanding of programming concepts, skills and a pre-questionnaire to see students' progress for group learning. Figure 2 shows that before the learning activities students get 100 minutes of learning about basic concepts and computer programming languages. They then took a pre-test to evaluate prior knowledge and took a self-efficacy questionnaire. After that, the lecturer introduced the nested if instruction and asked the students to practice working on the program. The troubleshooting activity lasts 100 minutes. During the learning activities students in the experimental group and control class were involved in completing two programming tasks. After the learning activities, all students took the posttest and filled out a questionnaire about self-efficacy.

Figure 2. The experimental procedure carried out

3.1. Measuring Instrument

The measuring instrument of the researcher refers to what is done by Wang and Hwang [6], namely the pretest and posttest of learning achievement of the two groups and a questionnaire for group learning self-efficacy and cognitive load.

Pretest evaluation was conducted to see that both groups had the same standards for programming knowledge and skills. Meanwhile, posttest is to see if there are differences between the two groups of the strategies applied.

Evaluation is done in the form of giving multiple choice questions in the form of pretest and posttest to test students’ programming concepts. The test consists of 50 multiple choice questions (70%) and is a single design question (30%). Multiple choice questions have been tested for validity. Multiple choice questions are to test students' knowledge of C++ programming while design questions to test problem solving programming skills. The questions given to
the students above were assessed based on a Likert Scale with five weighting criteria, namely: strongly agree (SS) with a weight of 5, agree (S) with a weight of 4, doubtful (RR) with weight 3, disagree (KS) with a weight of 2, and disagree (TS) with a weight of 1.

### 3.2. Analysis of Learning Achievement

ANCOVA analysis is used to see student’s achievement. As the dependent and independent variables, pretest and posttest scores are used. The aim is to see the significant difference in the achievement of the two groups.

The results of the homogeneity test of the regression coefficient of the pretest of the two groups were obtained $F = 0.000$, meaning that the homogeneity was not violated. and $p = 0.108 > 0.05$. Posttest was also carried out with ANCOVA as shown in Table 1.

To see students' achievement, a posttest was carried out in both groups with ANCOVA, a significant difference was obtained ($F = 1.24, p = 0.63 \not> 0.05$). Learning achievement increased compared to conventional collaborative learning with a squared value of $\eta^2 = 0.63$ in groups with problem-based learning strategies.

The questions for the posttest were tested for validity. Of the 50 items tested, there were 16 questions that were not valid questions. There are 34 valid questions out of 50 questions. Determination of the validity of the questions obtained from the calculation of the correlation coefficient ($\gamma_{pi}$). If the correlation of each factor is positive and the magnitude is 0.3 and above, then the factor is a strong construct and it can be concluded that the instrument items are valid. And vice versa, if the correlation value is below 0.3, it can be concluded that the instrument item is not valid.

The results of the homogeneity of variance with $F = 1.900$ ($p = 0.108 > 0.05$) showed no violation based on the homogeneity test of the knowledge and skills in pretest coefficient. This means that the knowledge and skills values from ANCOVA can be used.

Table 1 shows the results of the ANCOVA post test of learning achievement of the two groups.

<table>
<thead>
<tr>
<th>Difference</th>
<th>N</th>
<th>Means</th>
<th>SD</th>
<th>Adjusted mean</th>
<th>SE</th>
<th>F</th>
<th>$\eta^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instrument</td>
<td>15</td>
<td>77.82</td>
<td>5.84</td>
<td>77.82</td>
<td>.375</td>
<td>1.24</td>
<td>0.63</td>
</tr>
<tr>
<td>Control</td>
<td>16</td>
<td>71.30</td>
<td>14.08</td>
<td>71.30</td>
<td>.212</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### 3.3. Cognitive Load Analysis

Cognitive load testing with independent t Analysis for both groups was carried out as shown in Table 3. The average results obtained were 76.17 for the control class and 73.12 for the experimental class. It can be seen from the results of the two groups that the cognitive load of the control class is higher than the experimental class ($t = 3.620, p = 0.01 \not> 0.05$). The Cohen $d$ is 0.75, indicating a medium to large effect size.

Group study self-efficacy analysis

To see the significance of the self-efficacy of the two groups, ANCOVA analysis was performed where the pre-questionnaire was the covariance and the post-questionnaire was the dependent variable.

The results of the homogeneity of variance showed $F = 0.453$ with $p = 0.05$, which is not violating the homogeneity test of the regression coefficient on the pre-questionnaire of both groups. Post self-efficacy questionnaires were conducted for both groups with ANCOVA. Table 4 shows the average results, which were 7.23 and 7.01, meaning that the self-efficacy of both groups increased slightly ($F = 2.24, p = 0.05 \not> 0.05$). The squared value of $\eta^2 = 0.384$ represents a medium to large size. It can be concluded that the self-efficacy of the two groups for the experimental class is better in line with better learning achievement and lower cognitive load. Table 3 shows Independent sample t test of cognitive load from both groups and Table 4 shows the learning achievements of the two groups.

### Table 1. The results of the ANCOVA post-test of learning achievement of the two groups

<table>
<thead>
<tr>
<th>Difference</th>
<th>N</th>
<th>Means</th>
<th>SD</th>
<th>Adjusted mean</th>
<th>SE</th>
<th>F</th>
<th>$\eta^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instrument</td>
<td>15</td>
<td>77.82</td>
<td>5.84</td>
<td>77.82</td>
<td>.375</td>
<td>1.24</td>
<td>0.63</td>
</tr>
<tr>
<td>Control</td>
<td>16</td>
<td>71.30</td>
<td>14.08</td>
<td>71.30</td>
<td>.212</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table 2. The results of the ANCOVA post-test of learning achievement of the two groups

<table>
<thead>
<tr>
<th>Difference</th>
<th>Group</th>
<th>N</th>
<th>Means</th>
<th>SD</th>
<th>Adjusted mean</th>
<th>SE</th>
<th>F</th>
<th>$\eta^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Knowledge</td>
<td>Control</td>
<td>16</td>
<td>71.30</td>
<td>4.52</td>
<td>71.30</td>
<td>.982</td>
<td>2.24</td>
<td>0.63</td>
</tr>
<tr>
<td>Skills</td>
<td>Experiment</td>
<td>15</td>
<td>74.72</td>
<td>4.63</td>
<td>74.72</td>
<td>.964</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Knowledge</td>
<td>Control</td>
<td>16</td>
<td>70.61</td>
<td>3.32</td>
<td>70.60</td>
<td>.038</td>
<td>2.24</td>
<td>0.63</td>
</tr>
<tr>
<td>Skills</td>
<td>Experiment</td>
<td>15</td>
<td>77.62</td>
<td>2.05</td>
<td>77.62</td>
<td>.502</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Control 16 7.01 0.97 7.0 0.74

Learning media needs to be developed to increase the difficulty so as to reduce their cognitive load. A strategy group showed good learning achievement of the two groups, the problem-based strategy reduced the cognitive load which was much lower than that of the students in the treatment group. As for the self-efficacy of the two groups, the problem-based strategy group showed good learning achievement and lower cognitive load.

Cognitive processes in problem solving and knowledge construction with concept mapping facilitate cognitive processes. The use of internet technology can make collaboration and interaction easier in the learning process which has prompted many new collaborative learning environments that are being developed and investigated. The integration of social networking services within the learning environment has the potential to improve students’ achievement, interaction, collaboration and the overall experience. Learning assignments designed by students were adopted to replace the task of lecturers, to involve students in tasks with an appropriate level of difficulty so as to reduce their cognitive load. A learning media needs to be developed to increase the effectiveness and efficiency in learning.

Acknowledgements

The author would like to thank the Ministry of Education, Culture, Research and Technology of the Republic of Indonesia which has provided funds for this research through contract number 316/E4.1/AK.04.PT/2021 dated July 12, 2021 and LLDikti Region X Padang and the Institute of Business and Technology Pelita Indonesia involved in assisting this research.

Table 3. The results of the ANCOVA post-test of learning achievement of the two groups

<table>
<thead>
<tr>
<th>Group</th>
<th>N</th>
<th>Means</th>
<th>SD</th>
<th>t</th>
<th>p</th>
<th>d</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>15</td>
<td>76.17</td>
<td>1.779</td>
<td>3.620</td>
<td>0.01</td>
<td>1.7</td>
</tr>
<tr>
<td>Experiment</td>
<td>16</td>
<td>73.12</td>
<td>1.782</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4. The results of the ANCOVA post-test of learning achievement of the two groups

<table>
<thead>
<tr>
<th>Group</th>
<th>N</th>
<th>Means</th>
<th>SD</th>
<th>Average adjusted</th>
<th>SE</th>
<th>F</th>
<th>g^2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control</td>
<td>16</td>
<td>7.01</td>
<td>0.97</td>
<td>7.0</td>
<td>0.74</td>
<td>2.24</td>
<td>0.38</td>
</tr>
<tr>
<td>Experiment</td>
<td>16</td>
<td>7.23</td>
<td>0.92</td>
<td>7.26</td>
<td>0.63</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4. Conclusion and Recommendation

Collaborative learning with problem-based treatment with ANCOVA test results show an increase in learning achievement with the value of the square of Eta g^2 = 0.63. As for the cognitive load of the two classes, the average for the control group was 76.17 while for the treatment group it was 73.12. It means that the problem-based strategy reduced the cognitive load which was much lower than that of the students in the treatment group. As for the self-efficacy of the two groups, the problem-based strategy group showed good learning achievement and lower cognitive load.

Cognitive processes in problem solving and knowledge construction with concept mapping facilitate cognitive processes. The use of internet technology can make collaboration and interaction easier in the learning process which has prompted many new collaborative learning environments that are being developed and investigated. The integration of social networking services within the learning environment has the potential to improve students’ achievement, interaction, collaboration and the overall experience. Learning assignments designed by students were adopted to replace the task of lecturers, to involve students in tasks with an appropriate level of difficulty so as to reduce their cognitive load. A learning media needs to be developed to increase the effectiveness and efficiency in learning.

References


[4]. Ozurt, Ö. (2015). An Analysis on Distance Education Computer Programming Students’attitudes Regarding Programming and Their Self-Efficacy for Programming. Turkish Online Journal of Distance Education, 16(2), 111-121. DOI: 10.17718/tojde.58767.


