International Journal of Evaluation and Research in Education (IJERE)
Vol. 12, No. 2, June 2023, pp. 613~621
ISSN: 2252-8822, DOI: 10.11591/ijere.v12i2.23308

Student activities in solving mathematics problems with a computational thinking using Scratch

Neneng Aminah[1], Yohanes Leonardus Sukestiyarno[2], Adi Nur Cahyono[2], Siti Mistima Maat[3]
[1]Mathematics Education, Universitas Swadaya Gunung Jati, Cirebon, Indonesia
[2]Postgraduate Program of Mathematics Education, Universitas Negeri Semarang, Semarang, Indonesia
[3]Postgraduate Program of Mathematics Education, Universiti Kebangsaan Malaysia, Selangor, Malaysia

Article Info

Article history:

Received Nov 21, 2021
Revised Dec 9, 2022
Accepted Jan 6, 2023

Keywords:

Computational thinking
Mathematical problems
Problem solving
Student activities

ABSTRACT

The progress of the times requires students to be able to think quickly. Student activities in learning are always associated with technology and students' thinking activities and are expected to think computationally. Therefore, this study aimed to determine how learning with the concept of computational thinking (CT) using the Scratch program can improve students' mathematical problem-solving abilities. An exploratory research design was conducted by involving 132 grade VIII students in Kuningan, Indonesia. Data analysis began with organization, data description, and statistical testing. The results showed that students performed the concepts of abstraction thinking, algorithmic thinking, decomposition, and evaluation in solving mathematical problems. There were differences in students' problem-solving abilities before and after the intervention. Students' activeness in solving problems using the CT concept through a calculator significantly affected 52.3% of the ability to solve mathematical problems.

Corresponding Author:

Neneng Aminah
Mathematics Education, Universitas Swadaya Gunung Jati
Sunyaragi, Kesambi, Cirebon, West Java 45132, Indonesia
Email: nenengaminah255@gmail.com

1. INTRODUCTION

The rapid development of technology plays an important role in carrying out tasks in everyday life. To keep pace with technology, it is imperative that individuals have the education, knowledge, and skills to understand the technological systems they use and to be able to solve problems when things go wrong [1], [2]. Czerkawski and Lyman argued that individuals need to have the knowledge to respond to the challenges of the 21st century [1]. In education, it is necessary to pay attention to the curriculum to deal with this knowledge. Wing extends the notion of computing and proposes that computational thinking (CT) should be considered a fundamental skill taught throughout the curriculum [3]. As a developing country, Indonesia requires people who understand computer skills in terms of what they can and cannot do so that they can create effective computing tools. To achieve this, Indonesia must prepare the nation's successors to develop the knowledge they need to survive and effectively overcome the challenges of technological progress. One of the desired types of education is integrating technology into school subjects in the curriculum [4]–[6].

Technological developments are very influential on learning. The use of technology to complete tasks needs to be considered so that students can be involved in mathematical modeling activities [7]. The progress of times requires students to be able to think fast. Student activities in learning are always associated with technology and student thinking activities, and students are expected to be able to think computationally therefore, teachers must be able to design learning that leads to CT. Computational thinking is defined as a thought process for formulating and solving problems with the use of computers. Teaching computational thinking as a basic skill throughout the school curriculum will enable students to learn abstraction algorithmic, and logical thinking and solve complex, and open-ended problems [8]. The ability to solve problems is a level of ability that needs to be continuously implemented [9]. From the results of initial observations, it was found that students did not practice non-routine problems. Students feel confused when solving problems is one of the most typical characteristics of each individual. The problem-solving ability is accepted as an important cognitive activity in professional business and everyday life [10], [11].

In solving problems, one needs to go through several stages, including the thought process. Herskowitz et al. define abstraction as a vertical thinking activity of mathematical concepts previously built by the mind into a new mathematical structure [12]. At this point, the concept of CT contributes to problem-solving. Approaches to solving a problem, designing systems, and understanding human behavior that refers to the basic concepts of computing can use the concept of CT. The term CT is defined as a problem-solving approach that requires abstraction thinking, algorithmic decomposition, and pattern recognition [13]. However, in this study, we analyzed student activity using CT abstraction, decomposition, algorithm, and evaluation components [14], [15]. In this case, it is necessary to teach students how to solve problems during the educational process. Teaching activities are carried out to improve not only their problem-solving skills, but also explore computational thinking.

Learning that involves students directly in practice is highly coveted by students, for that class management is need in providing effective teaching. Behaviors related to effective classroom management need to be observed from the start to the end of learning on time: i) Managing lesson transitions; ii) Minimizing the time for things not related to assignments; iii) Handling wrong student behavior efficiently; iv) Preparing lessons well, and use appropriate learning media. Some learning media suitable for learning mathematics include the use of GeoGebra, Maple, Scratch, Tinkercad, Mapcitymath. Research showed that choosing the right media can improve students' cognitive abilities [16].

Students crave activities in the form of practices that make them active. Solving non-routine mathematical problems is a big problem faced by students today; this is an obstacle in learning. For this reason, we need a learning concept that can activate students through practice. Because the results of Singh's research [17] revealed that using cards in learning could be a medium that makes students practice and activities. However, to face the 21st century, it is better to use technological media to invite active students and think computers. Still, limited conditions after COVID-19 make teachers very careful in learning, so this research is very important to know activities when students solve problems with practical learning using computers. The concepts and practices used in CT involved computer science and other disciplines such as science, mathematics, social science, biology, art, language, and technique [18].

CT is increasing along with digital technology in every discipline [19] making it possible to be used to study other sciences, including mathematics. CT, which is defined as an approach to problem solving, is filled with problem solving abilities. CT in mathematics learning is being explored. The results of research in learning mathematics report the following things that are used as a reference for testing cognitive activity: i) Problem solving procedures are returned to students; ii) The teacher gives non-routine questions; iii) Asking problems that lead students to think more deeply; iv) Provide problems with small subsections; and v) Provide problems that can be solved in various ways [20].

Learning media is one of the supports to activate students [21], [22]. The media used must match the material to be studied. Students' use and technology skills have increased during the pandemic, but it is not yet certain whether they can increase their knowledge in exploring the material. After the pandemic took place, learning with health protocols was carried out properly, so it is time for researchers to take advantage of this situation for maximum use of technology. As research by Garneli and Chorianopoulos intervened in students' learning using video games, video game construction research results in projects with higher and more primitive CT skills, which were measured through project code analysis. In addition, the video game context further motivates students for future engagement with computational thinking activities [23].

The results of other studies show that Scratch can be used to develop students' mathematical ideas and computational thinking. Scratch's effect on the acquisition of mathematical concepts and the development of computational thinking [24]. The Scratch program can also build computational thinking. The study results also show that the trainees, in this case, elementary school teachers in Portugal, the use of the Scratch program can learn the concepts of computational thinking and develop useful products for practice in their classrooms [16], [25], [26]. From several research results that have been submitted, in this activity, researchers use a scratch program that is expected to improve computational thinking skills and students' problem-solving abilities in solving non-routine questions.

Increasing skills in using technology for students such as pandemic, but cognitively invisible. Therefore, this research provides a color difference in learning, especially mathematics with integrated technology, which uses the concept of CT thinking to improve the ability to solve problems. So, in this study, we analyze how student activities when learning to use technology in solving mathematical problems.

2. RESEARCH METHOD
2.1. Method and subject
This study uses a mixed method, using an exploratory sequential design, namely data collection begins with qualitative and then continues with quantitative data [27]. An exploratory research approach produces inductively derived generalizations about the group, process, activity, or situation under study. Next, these generalizations are assembled into a basic theory that explains the object of study [28]. This study was conducted on 132 students of eighth-grade of junior high school in Kuningan, Indonesia. Implemented post-pandemic, with due observance of health protocols. Interventions are limited class grouping. The data in this study were in the form of answer sheets for students' problem-solving ability tests in written form, which were analyzed using scoring guidelines with basic concepts of CT. Research instruments were in the form of CT ability test questions, observation sheets in videos, and interview guidelines in the form of audio. Interviews are used to explore processes that require clarity from test answer sheets and visible observations.

2.2. Data collection
The data in this study were in the form of answer sheets for students' problem-solving ability tests in written form, which were analyzed using an assessment guide with the basic concepts of CT. Observation data, and interview data. The research instruments were in the form of problem-solving ability test questions, observation sheets in the form of videos and field notes, and interview guidelines in audio form. Interviews are used to explore processes that require clarity from test answer sheets and visible observations.

2.3. Analyzing data
Researchers observed student activities during learning with the assistance of field observers. Activity notes were created, and student portfolios were collected. Debriefing sessions were conducted, and individual math problem-solving ability tests were also given before and after some activities. Student involvement was measured using a nominal data scale converted into interval data using the Method of Success (MSI) calculation. Qualitative data analysis began with organizing, annotating, and describing. For quantitative analysis using the SPSS version 22 program, data were collected from the results of student observations during learning when using the Scratch program, and also from students' problem-solving ability tests.

The research process was carried out in three stages. The first stage was an initial ability test. Then an intervention was carried out to see student activities when learning using the Scratch program and solving problems with the CT approach. Furthermore, an in-depth analysis of the activities was carried out by students. The triangulation stage was conducted through in-depth interviews to provide information from the test documents and visible observations. Statistical tests were conducted to see the success of activities and abilities before and after the intervention.

3. RESULTS AND DISCUSSION
3.1. Results
The research process began with testing the initial problem-solving ability of the sample to be studied. This activity was carried out to measure how the ability before doing computational thinking activities. From the initial test results, the minimum student score was 9, and the maximum student score was 25. After conducting the initial test, learning activities were carried out with the CT approach. In teaching, the teacher provided knowledge about using the Scratch program and giving non-routine questions. The teacher hopes that by continuing to practice making simple calculation programs, computing thinking will be built. Students were made accustomed to thinking in abstraction, algorithm, decomposition, or re-examining.

At the initial meeting, students were given exercises to recognize the buttons on the Starch program. Students were introduced to think abstractly from the problems given by the teacher. In this study, students were given problems about numbers in the material of sequences and series. The computer screen display is shown by Figure 1. The teacher paid attention to student activities to make simple programs, and this was meant to explore students' computational thinking processes to solve mathematical problems. Student activities are shown in Figure 1.

The first meeting did not explore computational thinking much, and students were explained how to understand the function of the buttons available from the Scratch program. From the first meeting, the students were enthusiastic to try out what they just knew. However, at the end of the activity, students were given the task of making a simple calculator, i.e., making multiplication, addition, division, and subtraction on number operations.

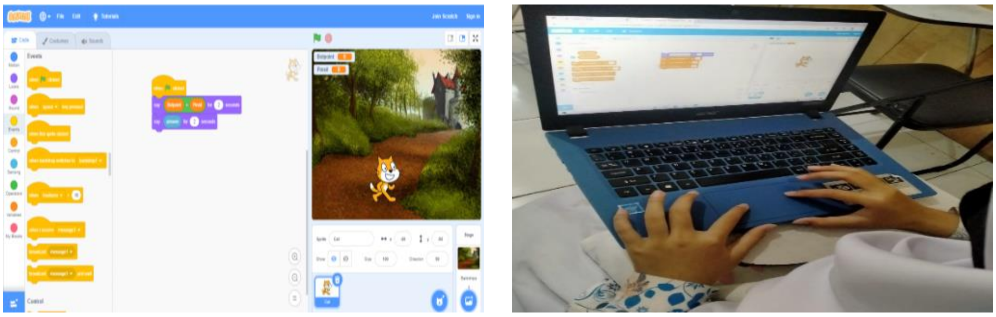


Figure 1. Examples of student activity while operating the Scratch button

The second meeting began by exploring the tasks that had been given in the first meeting. Students were asked the question, “Marcel bought a ballpoint pen and three pencils. How many items did Marcel buy? Make a mathematical sentence from the statement.” Almost all students think into a mathematical sentence in symbols, for example, a pencil and a pen. According to epistemology, this process is called abstraction. The occurrence of abstraction goes from an unstructured abstraction to a developing abstraction [29].

The increase in activities carried out by students is increasingly visible abstraction thinking begins to run. Students start the process of thinking from what they imagine into a symbol. During the activity of making a simple calculation, students repeat their previous learning. This iteration process or algorithm begins with reading, understanding, conveying to the brain, then proceeding to develop thoughts poured into the writing they make through scribbles on the computer. The thinking activity carried out by students is an abstraction process, while the result of the process is an image that is thought to be a concept. The process of abstraction is an activity, while the result of the abstraction is a concept [30]. Figure 2 shows the examples of student activities when they began to think computationally.

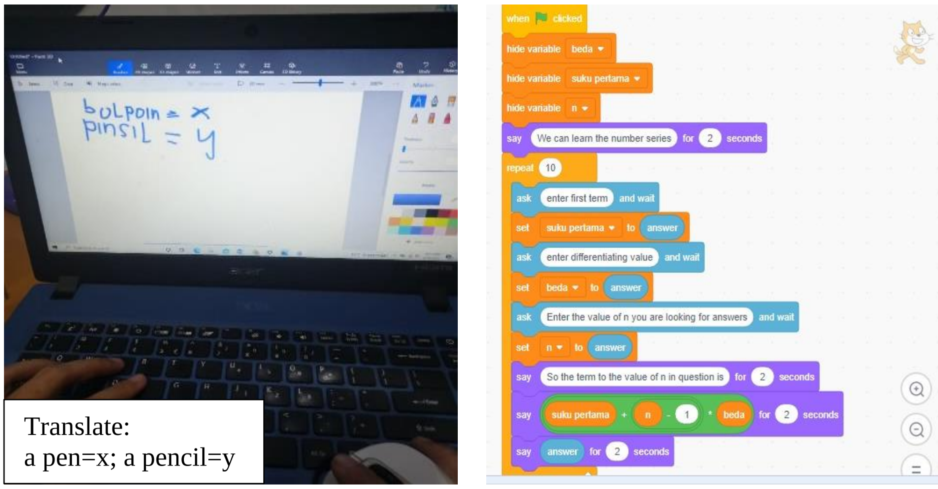
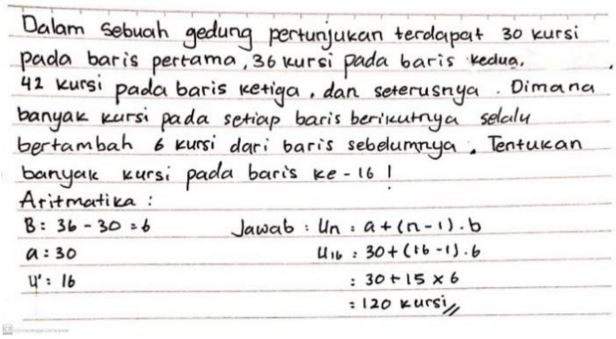


Figure 2. Examples of student’s scribble during abstraction thinking and algorithm thinking

At the next stage, students made plans to solve problems with a step-by-step thinking process. This process in computational thinking is called algorithmic thinking. Algorithmic views in life are valuable because they involve many important activities by following simple and separate steps [8], [29], [31], [32]. The algorithm is a skill to design a series of operations involving a regular sequence of steps in solving a problem or completing a task using computational operations. However, few students also experienced errors when making a simple calculation, as shown in Figure 2.

In the third meeting, the teacher gave problems about number lines and series. In the first stage, the students were given the activity of making a simple calculation for the nth term, i.e., making simple operations using Scratch, $Un = a + (n - 1)b$, then they were given a math problem. Students were getting used to using the Scratch program. Students were trained to think computationally to solve number series problems. From the results of the test answers, the mindset of students can be observed. Then, to verify whether what they were doing was correct, students used a simple calculation that they have made. The following is an example of a student's answer in solving mathematical problems by CT.


Dalam sebuah gedung pertunjukan terdapat 30 kursi pada baris pertama, 36 kursi pada baris kedua, 42 kursi pada baris ketiga, dan seterusnya. Dimana banyak kursi pada setiap baris berikutnya selalu bertambah 6 kursi dari baris sebelumnya. Tentukan banyak kursi pada baris ke-16!

Aritmatika:

B: 36 - 30 = 6 Jawab: Un = a + (n-1) . b

a: 30 U16 : 30 + (16-1) . 6

U: 16 : 30 + 15 × 6

: 120 kursi

Translation:
In a theatre, there are 30 seats in the first row, 36 seats in the second row, 42 seats in the third row, and so on, where the number of subsequent seats always increases by six seats from the previous row. Determine the number of seats in the 16th row.

Arithmetic: $b = 36 - 30 = 6$; $a = 30$; $u = 16$.
answer:
$Un = a + (n-1)b$
$U_{16} = 30 + (16-1)6$
$U_{16} = 30 + (15 \times 6) = 120\ chairs$

Figure 3. Examples of student problem-solving skills test results

Students used the problem-solving stage when solving problems, as expressed according to Polya [33] through the stages of understanding the problem, making plans, implementing plans, and looking back. In line with the stages of problem-solving from Krulik and Rudnick [34], several stages of reading problems, exploring problems, choosing strategies, solving problems, and reviewing were revealed. Meanwhile, according to Dewey, the problem-solving step begins with facing the problem, defining it, finding solutions, conjecturing consequences, and finally testing the consequences.

In the first stage of problem-solving, students were trained continuously to carry out the abstraction thinking process, where the result of the process was an image that was thought to be a concept. Suggested that the abstraction process is an activity, while the result of the abstraction is a concept. Figure 3 shows the abstraction process of students when writing b=6, a=30, and U=16 here. There was a slight error when writing U=16. It should be n=16 or the 16th order. From here, data in the form of interviews were taken, one of the students stated that "*I symbolize the first order with a=30, then from the second-order minus the first order, it produces a difference. I symbolize it with b=36-30, then b=6 and U=16 means the sixteenth order*." The presentation provides information that students have carried out an abstraction thinking process. The result of the process was a picture of what they were thinking with the result of a symbol that defines the problem given.

The second stage in problem-solving was making plans, exploring problems, and defining problems. Although it is a little different, the goal was to explore students to compile problems, and accidentally students have done decomposition thinking. The thinking process divided big problems into small problems in order to facilitate the thinking process. The concept of computational thinking was called decomposition. Decomposition can break down complex tasks (problems) into smaller more detailed tasks [35]. Not only seen in the test answers, in the activity of making simple calculations, students also began to collect buttons and which operations would be used to make sequence formulas on number series material.

In the third stage in problem-solving, students must solve problems. In this stage, students were already moving in the process of providing stages to solve problems. This coincides with the concept of computational thinking. Students begin to think step by step. This process is usually called algorithmic thinking. Algorithms in life are considered very important because this process involves many activities by following simple and separate steps [36]. An algorithm is a skill to design a series of operations involving the use of a sequence of steps regularly in solving problems or completing tasks, using calculation operations if seen from Figure 3 this process is at the time of compiling calculations from the formula Un=a+(n-1)b=30 +(16-1)6=120 in the 16th with 120 seats.

The final stage was reviewing. The teacher reviewed students trying calculation activities with the help of the Scratch they made. Activities carried out after the abstraction process using symbols and known values were substituted in the program. When there was an input command for the first term, students entered the number 30. The next command inputs different values because previously, they had gone through the stages of abstraction and decomposition thinking. Students entered the number 6, then entered the requested sequence. The student entered the number 16, then the program proceeded, and the answer showed 120.

Students' activities when answering math problems were included in the answers. Students carried out computational thinking processes through analysis. It seemed like abstract thinking, do algorithm when making steps, doing decomposition thinking, and getting to the stage of re-examining the Scratch program that they made as a calculating tool in verifying the answers they found. The findings show that the mean learning activity scores varied for the four sub-indicators, ranging from 1.00 to 5.00. A standard deviation of 3.50 indicates adequate variability across all sub-indicators. This value indicates that students' activities to carry out these tasks vary. Students were very enthusiastic about participating in activities. The previous nominal scale data was changed using the MSI formula into interval data.

In general, students could solve problems given by using the concept of computational thinking. The average percentage of computational thinking during learning using the scratch program was 34.57 at the first meeting; the activities carried out are still easy, and the introduction of scratch buttons. In the second meeting, the average decreased to 31.34 because the level of difficulty increased. However, the third meeting increased again at 33.09 because the teacher reminded the students from beginning to end. Students took the initial and final tests of problem-solving abilities. Both tests were used to measure problem-solving skills with computational thinking. Furthermore, to determine the effect of the intervention given through activities during learning activities using the Scratch program on problem-solving abilities, statistical analysis was carried out on student scores before and after the intervention. The previous nominal scale data was changed using the MSI formula into interval data. A comparative test using SPSS version 22 showed that group of data was normally distributed. A comparative test showed that one group of data was normally distributed. Kolmogorov Smirnov normality test showed that the students' scores were normally distributed $p>0.05$; The test homogeneity $p>0.05$ were data homogeny.

The analysis was continued using the one-way ANOVA with the findings that there were significant differences in each meeting with a value of sig=0.000 with F value $21.47>F_{tabel(2,85).}$ The post hoc test showed there was no significant difference from the first to the third activity. From the observations, the first meeting was still introducing scratch buttons, the second meeting activity looked a little complicated for 14-year-old students, continued at the third meeting, the students were familiar with buttons and commands to continue to think computationally. For the accuracy of the data, the researcher got a statement from the interview results of one of the subjects, "*at the first meeting we were easy to follow the learning, the second meeting we had to think hard to get the formula to be used, at the third meeting we were used to using this program*." To see the success of the learning activity intervention, the researcher continued on the problem-solving ability test before and after the intervention.

The average problem-solving ability of students after the intervention 83.22, was higher than the problem-solving ability before the intervention 15.41. Standard deviation 4.858 and 4.528 shows adequate variability across all variable. Then the repaired test was carried out. The t-test was used to analyze the data further paired sample t-test. The results of the t-test showed a significant difference ($p<0.05$) between students' problem-solving abilities before and after the intervention. Students' activities also influence students' mathematical problem-solving abilities in learning using Scratch. Next, do a test analysis of the influence of students' computational thinking activities with problem solving abilities. The results of the regression test showed that there was a significant positive effect ($p<0.05$) on student activity (x) on mathematical problem-solving skills using the concept of students' computational thinking (y). This relationship is shown by the regression equation Y=3,671+1,662X. The value of the coefficient of the independent variable is positive. Every time there is an increase in the activity of each 1 unit, it will increase the ability to solve mathematical problems by thinking computationally by the number of coefficients multiplied by the respective independent variable units. Student activities contributed 52.3%. This shows that learning activities with CT activities through the implementation of the Scratch program have succeeded in influencing mathematical problem-solving abilities.

3.2. Discussion

Class VIII students of the subjects selected in this study have already carried out learning activities using the Scratch program. The activity shown by the participants is seen when carrying out the abstraction process, the way of thinking is poured by giving symbols. In the activities carried out during the manufacture of simple calculations and thinking activities that were reflected in answering questions, students created and projected their abilities on a new concept, i.e., changing the problems they read into mathematical symbols. Piaget put forward the concept of reflective abstraction based on the coordination of relations and object operations. The reflective abstraction proposed by Piaget [37], which focuses on ideas about action and operation, is the same, i.e., abstraction does not use the sensory-motor or material objects. According to Glasersfeld [38], reflective abstraction refers to the subject's ability to project and reorganize the structure created from the subject's activities and interpretations to a new level.

In the first stage in planning problem solving, all participants carried out procedural activities starting from reading, understanding the problem, and then pouring it into a mathematical symbol. At the secondary school level, it is included in the category of abstraction that is a reflective structural abstraction as thinking that develops through awareness from a different point of view from the faced problem [39]. In this abstraction, students can demonstrate the ability to anticipate the results of problem-solving and provide arguments for their decisions. Students are able to reflect on the decisions obtained for the next activity. They are also able to symbolize clearly to arrive at the final destination.

The second stage was compiling the problems they face. Participants carry out step-by-step activities for the flow of thoughts to solve problems. This activity is called algorithmic thinking [4], [5]. In line with the opinion of Sys aspects of algorithmic thinking in computational thinking are key competencies to be mastered at the high school level and academic informatics and ICT studies [36]. Algorithms are skills to string together a series of operations or actions step by step to solve problems. Participants also think about turning big problems into small problems. This is what they do when solving problems, and this thinking is called decomposition [40]. In the stages of re-examining activities, from the results of observations, it was seen that students tried the simple calculations made to ensure that they were appropriate or not from the answers to the problem-solving abilities they were doing by utilizing technology. Everything that the participants did in the final stage means that they had carried out a method of confirming, rechecking, and confirming. Carry out an evaluation process from the experience of the previous stages.

The findings show that during making a simple calculator or answering problem-solving skills, the subject uses the concept of computing thinking that strengthens the CT abstraction, decomposition, and algorithm [31]. While many researchers propose other concepts, Barr and Stephenson added automation thinking to the CT concept [5]. Likewise, Ioannou and Angeli added debugging and generalization [4]. Meanwhile, Selby added an evaluation and thought of generalization [41]. The main CT model defines problems, solves problems, and analyzes solutions [5], [8]. Previous research discussed learning with technology using scratch and aia contributing so that students significantly improve their performance the criteria for evaluating attitudes and knowledge in programming. However, in contrast to these research findings, research conducted in the field of programming [42] shows that learning uses technology. In particular, the scratch program significantly contributes to the ability to solve mathematical problems. Students feel happy, respect more learning, and have a happy attitude toward mathematics. This study produced that problem solving can use the computing thought approach carried out in high school students, slightly different from research [43] who researched the use of scratch programs in early age students. But both of these studies together produced fun teaching for students.

4. CONCLUSION

Based on the introduction, data analysis, and discussion, this study shows that learning CT through the Scratch program can explore CT skills. From these findings, the concept of CT can be used to solve problems in learning mathematics. From these findings, it can be seen that students actively learn and express their mathematical ideas through the thinking process they go through when making a simple calculator using the scratch program. This study provides recommendations for solving mathematical problems using abstract thinking concepts algorithms and decomposition and evaluation. Another research finding is a significant difference in students' problem-solving abilities before and after the intervention. Student activity through making a simple calculator in the scratch program contributed 52.3% to solving mathematical problems.

The results of this study, the concepts found can be used as a reference for teachers in designing lesson plans, learning media, and learning strategies used to continue the CT process in solving math problems. In addition, for further research, it is essential to examine learning that can measure CT concept skills through the implementation of other mathematics programs, such as the use of the GeoGebra application and habituation by giving non-routine questions.

ACKNOWLEDGEMENTS

The authors would like to thank to Universitas Swadaya Gunung Jati, Indoneisa, Universitas Negeri Semarang, Indonesia, Universitas Kebangsaan Malaysia, Malaysia, and SMP Negeri in Kuningan, Indonesia.

REFERENCES

[1] B. C. Czerkawski and E. W. Lyman, "Exploring Issues About Computational Thinking in Higher Education," *TechTrends*, vol. 59, no. 2, pp. 57–65, Mar. 2015, doi: 10.1007/s11528-015-0840-3.

[2] J. M. Wing, "Computational thinking and thinking about computing," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 366, no. 1881, pp. 3717–3725, Oct. 2008, doi: 10.1098/rsta.2008.0118.

[3] J. Togyer and J. M. Wing, "Computational Thinking: What and Why?" the Link - The Magazine of the Carnegie Mellon University School of Computer Science, 2010. [Online]. Available: https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why
[4] I. Ioannou and C. Angeli, "A Framework and an Instructional Design Model for the Development of Students' Computational and Algorithmic Thinking," *Mediterranean Conference on Information Systems (MCIS)*, 2016, pp. 1–8.
[5] V. Barr and C. Stephenson, "Bringing computational thinking to K-12," *ACM Inroads*, vol. 2, no. 1, pp. 48–54, Feb. 2011, doi: 10.1145/1929887.1929905.
[6] A. Yadav, H. Hong, and C. Stephenson, "Computational Thinking for All: Pedagogical Approaches to Embedding 21st Century Problem Solving in K-12 Classrooms," *TechTrends*, vol. 60, no. 6, pp. 565–568, Nov. 2016, doi: 10.1007/s11528-016-0087-7.
[7] A. N. Cahyono, Y. L. Sukestiyarno, M. Asikin, Miftahudin, M. G. K. Ahsan, and M. Ludwig, "Learning mathematical modelling with augmented reality mobile math trails program: How can it work?" *Journal on Mathematics Education*, vol. 11, no. 2, pp. 181–192, 2020, doi: 10.22342/jme.11.2.10729.181-192.
[8] S. Cynthia, "Computational Thinking: The Developing Definition," in *ITiCSE Conference 2013*, 2013, pp. 5–8.
[9] I. Wahyuni, N. Aminah, Y. L. Sukestiyarno, and A. Wijayanto, "Development of evaluation of mathematical communication capabilities based on information technology for junior high school students," *Journal of Physics: Conference Series*, vol. 1470, no. 1, p. 012048, Feb. 2020, doi: 10.1088/1742-6596/1470/1/012048.
[10] M. Özenç and C. Çarkit, "The relationship between functional literacy and problem-solving skills: A study on 4th-grade students," *Participatory Educational Research*, vol. 8, no. 3, pp. 372–384, Aug. 2021, doi: 10.17275/per.21.71.8.3.
[11] I. Wahyuni *et al.*, "Design of instrument Technological Pedagogic Content Knowledge (TPACK) for prospective mathematics teachers," *Journal of Physics: Conference Series*, vol. 1918, no. 4, p. 042097, Jun. 2021, doi: 10.1088/1742-6596/1918/4/042097.
[12] M. C. Mitchelmore and P. White, "Development of angle concepts by progressive," *Educational Studies in Mathematics*, vol. 41, pp. 209–238, 2000.
[13] E. C. Bouck, P. Sands, H. Long, and A. Yadav, "Preparing Special Education Preservice Teachers to Teach Computational Thinking and Computer Science in Mathematics," *Teacher Education and Special Education: The Journal of the Teacher Education Division of the Council for Exceptional Children*, vol. 44, no. 3, pp. 221–238, Aug. 2021, doi: 10.1177/0888406421992376.
[14] N. Aminah, Y. Sukestiyamo, Wardono, and A. N. Cahyono, "A Teaching Practice Design Based on a Computational Thinking Approach for Prospective Math Teachers Using Ed-Tech Apps," *International Journal of Interactive Mobile Technologies*, vol. 16, no. 14, pp. 43–62, 2022, doi: 10.3991/ijim.v16i14.30463.
[15] N. Aminah, Y. Leonardus, W. Wardono, and A. Nur, "Computational Thinking Process of Prospective Mathematics Teacher in Solving Diophantine Linear Equation Problems," *European Journal of Educational Research*, vol. 11, no. 3, pp. 1495–1507, Jul. 2022, doi: 10.12973/eu-jer.11.3.1495.
[16] M. J. Marcelino, T. Pessoa, C. Vieira, T. Salvador, and A. J. Mendes, "Learning Computational Thinking and scratch at distance," *Computers in Human Behavior*, vol. 80, pp. 470–477, Mar. 2018, doi: 10.1016/j.chb.2017.09.025.
[17] P. Singh, T. S. Hoon, A. Md Nasir, A. Md Ramly, S. Md Rasid, and C. C. Meng, "Card game as a pedagogical tool for numeracy skills development," *International Journal of Evaluation and Research in Education (IJERE)*, vol. 10, no. 2, pp. 693–705, Jun. 2021, doi: 10.11591/ijere.v10i2.20722.
[18] G. Gadanidis, R. Cendros, L. Floyd, and I. Namukasa, "Computational thinking in mathematics teacher education," *Contemporary Issues in Technology & Teacher Education*, vol. 17, no. 4, pp. 458–477, 2017.
[19] C. (Yu) Pei, D. Weintrop, and U. Wilensky, "Cultivating Computational Thinking Practices and Mathematical Habits of Mind in Lattice Land," *Mathematical Thinking and Learning*, vol. 20, no. 1, pp. 75–89, Jan. 2018, doi: 10.1080/10986065.2018.1403543.
[20] W. K. Ho, C. K. Looi, W. Huang, P. Seow, and L. Wu, "Realizing Computational Thinking in the Mathematics Classroom: Bridging the Theory-Practice Gap," *Proceedings of the 24th Asian Technology Conference in Mathematics*, 2019, pp. 35–49.
[21] N. Aminah and I. Wahyuni, "The ability of pedagogic content knowledge (PCK) of mathematics teacher candidate based on multiple intelligent," *Journal of Physics: Conference Series*, vol. 1280, no. 4, p. 042050, Nov. 2019, doi: 10.1088/1742-6596/1280/4/042050.
[22] N. Aminah and I. Wahyuni, "Pedagogic Content Knowledge (PCK) Ability of Prospective Mathematics Teachers in the Field Experience Program at the Cirebon City State Junior High School/Senior High School," *Jurnal Nasional Pendidikan Matematika*, vol. 2, no. 2, pp. 259–267, 2018.
[23] V. Garneli and K. Chorianopoulos, "Programming video games and simulations in science education: exploring computational thinking through code analysis," *Interactive Learning Environments*, vol. 26, no. 3, pp. 386–401, Apr. 2018, doi: 10.1080/10494820.2017.1337036.
[24] J. A. Rodríguez-Martínez, J. A. González-Calero, and J. M. Sáez-López, "Computational thinking and mathematics using Scratch: an experiment with sixth-grade students," *Interactive Learning Environments*, vol. 28, no. 3, pp. 316–327, Apr. 2020, doi: 10.1080/10494820.2019.1612448.
[25] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning Face Representation from Scratch," *Computer Vision and Pattern Recognition*, 2014, doi: 10.48550/arXiv.1411.7923.
[26] T. Ferrer-Mico, M. À. Prats-Fernàndez, and A. Redo-Sanchez, "Impact of Scratch Programming on Students' Understanding of Their Own Learning Process," *Procedia - Social and Behavioral Sciences*, vol. 46, pp. 1219–1223, 2012, doi: 10.1016/j.sbspro.2012.05.278.
[27] Y. Sukestiyarno, *Research Educational Method*. Semarang: UNNES Press, 2020.
[28] R. Stebbins, *Exploratory Research in the Social Sciences*. SAGE Publications, Inc., 2001.
[29] P. J. Rich, G. Egan, and J. Ellsworth, "A Framework for Decomposition in Computational Thinking," in *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, Jul. 2019, pp. 416–421, doi: 10.1145/3304221.3319793.
[30] I. Cetin and E. Dubinsky, "Reflective abstraction in computational thinking," *The Journal of Mathematical Behavior*, vol. 47, pp. 70–80, Sep. 2017, doi: 10.1016/j.jmathb.2017.06.004.
[31] C.-K. Looi, M.-L. How, W. Longkai, P. Seow, and L. Liu, "Analysis of linkages between an unplugged activity and the development of computational thinking," *Computer Science Education*, vol. 28, no. 3, pp. 255–279, Jul. 2018, doi: 10.1080/08993408.2018.1533297.
[32] F. K. Pala and P. Mıhcı Türker, "The effects of different programming trainings on the computational thinking skills," *Interactive Learning Environments*, vol. 29, no. 7, pp. 1090–1100, Oct. 2021, doi: 10.1080/10494820.2019.1635495.
[33] G. Polya, *How to Solve It: A New Aspect of Mathematical Method*, 2nd ed. New Jersey: Princeton University Press, 1973.

[34] S. Klurik and J. A. Rudnick, *The New Sourcebook for Teaching Reasoning and problem Solving in Elementary School*, 1st ed. Massachusetts: Schuster Company, 1993.

[35] J. M. Wing, "Computational thinking's influence on research and education for all," *Italian Journal of Educational Technology*, vol. 25, no. 2, pp. 7–14, 2017, doi: 10.17471/2499-4324/922.

[36] M. M. Sysło, "From Algorithmic to Computational Thinking," *ITiCSE '15: Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*, Jun. 2015, doi: 10.1145/2729094.2742582.

[37] J. Piaget, *The Construction of Reality in The Child*, 1st ed. London: Routledge, 2013.

[38] V. Glasersfeld, *Abstraction, re-presentation, and reflection*. New York: Springer Verlag, 1991.

[39] E. Gray and D. Tall, "Abstraction as a natural process of mental compression," *Mathematics Education Research Journal*, vol. 19, no. 2, pp. 23–40, Sep. 2007, doi: 10.1007/BF03217454.

[40] C. C. Selby, "Relationships: computational thinking, pedagogy of programming, and Bloom' s Taxonomy," in *Proceedings of the Workshop in Primary and Secondary Computing Education*, Nov. 2015, pp. 80–87, doi: 10.1145/2818314.2818315.

[41] D. Phillips and J. Woollard, "The developing concept of 'computational thinking'," 2016. [Online]. Available: https://www.researchgate.net/publication/316083867_The_developing_concept_of_computational_thinking

[42] N. Zaranis, V. Orfanakis, S. Papadakis, and M. Kalogiannakis, "Using Scratch and App Inventor for teaching introductory programming in Secondary Education. A case study," *International Journal of Technology Enhanced Learning*, vol. 1, no. 1, p. 1, 2016, doi: 10.1504/IJTEL.2016.10001505.

[43] S. Papadakis, M. Kalogiannakis, and N. Zaranis, "Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: a case study," *International Journal of Mobile Learning and Organisation*, vol. 10, no. 3, pp. 187–202, 2016, doi: 10.1504/IJMLO.2016.077867.

BIOGRAPHIES OF AUTHORS

Neneng Aminah received the Doctor degree in mathematics education from the State University of Semarang, Indonesia. She is a permanent lecturer in the Mathematics Education Study Program at Swadaya Gunung Jati University since 2008. Her teaching experiences include teaching Basic Statistic in Education, Entrepreneur, Teaching Practice. She can be contacted at email: nenengaminah255@gmail.com.

Yohanes Leonardus Sukestiyarno is a civil servant lecturer at Semarang State University. Currently, he is not in further study. He joined the State University of Semarang as a civil servant lecturer on March 1, 1984. His last education was doctoral and graduated on November 14, 1999. He is a civil servant Lecturer with the position of Professor and Head of the Doctoral Study Program Mathematics Education Semarang State University. He teaches the main subjects of Research Methods; Statistics; probability theory; Computer 1 & 2; Science philosophy; Learning mathematics in English. He can be contacted at email: sukestiyarno@mail.unnes.ac.id.

Adi Nur Cahyono is a lecturer in Mathematics Education at the Department of Mathematics, Universitas Negeri Semarang (UNNES), Indonesia since 2008. He is also an Expert Staff of the Rector of UNNES for Academic Affairs in 2019-2021. In 2017, he received his doctoral degree (Dr.rer.nat.) from J.W.v. GoetheUniversität Frankfurt am Main, Germany, in the specialty of Didactics of Mathematics for his work on the MathCityMap Indonesia. He teaches and supervises undergraduate, masters, and doctoral students in geometry courses, teaching & learning mathematical modelling, and mathematics education media. He is the founder of the Mobile Math Trails Research Group (mathe.id), where he and his team conduct research and work with a focus on math trails, mathematical modelling, and mathematics education with digital technologies, in collaboration with teachers and researchers from other institutions both in Indonesia and abroad. He can be contacted at email: adinurcahyono@mail.unnes.ac.id.

Siti Mistima Maat is an Associate Professor at the Faculty of Education, Universiti Kebangsaan Malaysia. She obtained her Ph.D in Mathematics Education from Universiti Kebangsaan Malaysia after receiving her M.Sc in Mathematics from Universiti Putra Malaysia and B.Sc (Hons) in Mathematics, from Universiti Pertanian Malaysia. She has more than 25 years' experience in teaching mathematics for students from private and local education institutions. She has authored more than 100 publications including journal articles, books and proceedings. Her research interests include mathematics teacher knowledge, teacher professional development, advanced statistical analysis and history of mathematics. She can be contacted at email: sitimistima@ukm.edu.my.