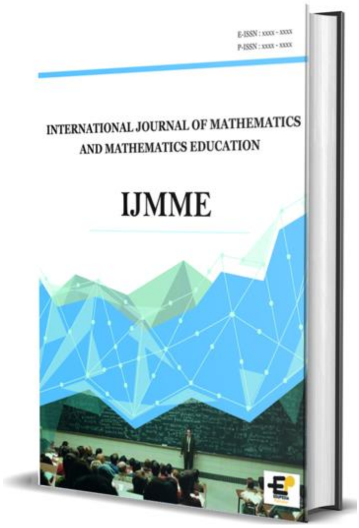

https://journals.eduped.org/index.php/IJMME

Students' Proficiency in Computational Thinking Through Constructivist Learning Theory

Lilis Marina Angraini*

Universitas Islam Riau, Indonesia

Nia Kania

Universitas Majalengka, Indonesia

Ferit Gürbüz

Kırklareli University, Türkiye

To cite this article:
Angraini, Kania & Gürbüz. (2024). Students' Proficiency in Computational Thinking Through Constructivist Learning Theory. *International Journal of Mathematics and Mathematics Education (IJMME), 2*(1), 45-59 https://doi.org/10.56855/ijmme.v2i1.963

IJMME International Journal of Mathematics and Mathematics Education
February 2024, Vol. 02, No. 01, 45-59 doi: 10.56855/ijmme.v2i1.963

Students' Proficiency in Computational Thinking Through Constructivist Learning Theory

Lilis Marina Angraini*, **Nia Kania**, **Ferit Gürbüz**

Article Info

Article History

Received:
November 27, 2023
Accepted:
December 27, 2023

Keywords

Computational Thinking
Mathematics Education
Algebraic Structures

Abstract

This study aims to analyse the mathematical computational thinking ability of Mathematics Education study program students in terms of constructivism learning theory. The main focus of this research is to understand how the constructivist learning approach affects the development of students' mathematical computational thinking ability. This research is qualitative descriptive research. It explores students' learning process in the context of developing computational understanding and application of computational concepts in mathematics education. This research was conducted in the Department of Mathematics Education on students who took the Algebraic Structure course in the 2023/2024 academic year. The subjects of this research amounted to 34 students. The findings of this study provide insight into the effectiveness of constructivism learning theory in improving the mathematical computational thinking skills of Mathematics Education students. The results showed that students' average mathematical computational thinking ability was good, and descriptively, the prior mathematical knowledge could also differentiate students' mathematical computational thinking ability in terms of constructivism learning theory.

Introduction

In an era where information and communication technologies increasingly dominate everyday life, using computing in mathematics education is becoming increasingly important. Technological developments have changed how we interact with mathematics, and mathematics education must adapt to this challenge. Amidst these developments, the importance of computational thinking skills must be addressed (Ye et al., 2023; Barr & Stephenson, 2011; Voogt et al., 2015).

Computational thinking skills involve understanding basic computing concepts, including problem-solving, algorithms, programming, and data. These essential skills can help students solve mathematical problems effectively, understand real-world applications of mathematics, and prepare for an increasingly technology-dependent future (Grover & Pea, 2013; Kale et al., 2018).

Mathematics Education is essential in developing students' computational thinking skills, especially Mathematics Education students. They are expected to be agents of change in teaching mathematics by integrating technology and computation. Therefore, it is essential to understand the extent to which Mathematics Education students have developed computational thinking skills.

Learning approaches based on constructivism learning theory strongly relate to developing computational thinking skills. Constructivism emphasises student-centred learning, where students build their understanding through active experience and reflection. In this context, research on the computational thinking skills of Mathematics Education students with a constructivist perspective is critical (Suprapti, 2018; Saputro & Pakpahan, 2021; Sari & Kurniawan, 2023).

The critical role of Mathematics Education students in developing computational thinking skills is very significant, and understanding computational concepts in the context of mathematics education is essential. Mathematics Education students are expected to be agents of change in technology and computing in mathematics learning in schools. They should be able to integrate computational tools, such as math software and programming, into the mathematics learning process to facilitate students' understanding of concepts and the development of computational thinking skills (Brown & Capper, 2019; Selden et al., 2014).

Mathematics Education students will often become mathematics teachers in schools. Therefore, they should be able to guide students in developing deep mathematical understanding and computational thinking skills. These students will play an essential role in helping students overcome difficulties in understanding mathematical concepts through computational approaches. Mathematics Education students can play a role in developing computational-based learning materials, such as interactive math games, simulations, or math applications. This can help students internalise mathematical concepts more interestingly and effectively.

Computational thinking skills help students in developing creative thinking and problem-solving skills. Mathematics Education students should understand how computation can be used to formulate and solve mathematical problems. They can encourage students to think computationally and seek innovative solutions. Computational thinking skills are relevant not only for students but also for teachers. Mathematics Education students will continue to learn and develop in their profession. Therefore, computational understanding can help them continue to learn and utilise technology in mathematics teaching (Lee & Yadav, 2016; Ye et al., 2023).

Constructivism Learning Theory is a framework in education that emphasises the active role of students in constructing their knowledge through experience, reflection and interaction with learning materials. This theory highlights student-centered learning and an emphasis on deep understanding. Constructivism emphasises that learning should not just be the delivery of information from lecturer to student but involve students actively. This means that lecturers act as facilitators, assisting students in understanding mathematical concepts by supporting exploration and problem-solving (Baran & Maskan, 2021; Yılmaz, 2017).

Constructivism theory says students construct their knowledge by formulating meaning from learning experiences. They create mental representations and concepts based on their interaction with learning materials (Duffy & Jonassen, 2013; Akkoç & Ocak, 2019). In this context, students who understand computing concepts can develop their understanding through active exploration. This theory emphasises the importance of collaboration and social interaction. Students learn by discussing, sharing and collaborating with fellow students. In the context of computational understanding, students can discuss and work together to solve computational problems.

Constructivism theory encourages the development of critical thinking skills. Students are encouraged to question, analyse and formulate their understanding. Students should think critically about algorithms, logic, and problem-solving in terms of computational understanding. This theory emphasises the importance of contextualising learning. Students should be able to see the relevance and applicability of mathematical concepts in real-world situations. In computational understanding, contextualisation helps students connect computational concepts with everyday applications.

The constructivist approach is very relevant in developing computational thinking skills. Constructivism requires students to be active in their learning. In understanding computational concepts, students need to explore, try, and face computational problems. This approach allows students to actively design algorithms, run code, and face programming challenges. Constructivism theory encourages students to collaborate and discuss. In the context of computational understanding, students can work together to solve problems and share knowledge. Exchanging ideas and solutions can enrich their understanding (Grover & Pea, 2013; Liu et al., 2021).

Constructivism encourages the development of critical thinking skills. Students must think logically, analyse problems, and formulate effective solutions to understand computing. The constructivism approach helps students understand the logic and algorithms underlying computing. This theory encourages deep understanding rather than mere memorisation. To understand computing, students need to understand the basic principles and concepts of computing. Constructivism allows students to formulate solid knowledge through exploration and practice. Constructivism emphasises contextualising learning in real situations. In computational understanding, students can relate computational concepts to real-world applications, such as application development, problem-solving, and data analysis (Kale et al., 2018; Yılmaz, 2017).

Algebraic structures courses are part of the mathematics education curriculum that cover higher topics in abstract algebra. It may include group theory, rings, fields, and other complex algebraic topics. Constructivism learning theory is a theoretical approach emphasising that students construct knowledge through personal experiences. It creates a close relationship between the learning process and students' formation of meaningful knowledge.

This study aims to investigate the computational thinking ability of Mathematics Education students by focusing on the framework of constructivism learning theory. This research seeks to understand how constructivism learning theory can develop students' computational thinking ability and how student-centred learning processes and knowledge construction can influence the understanding of computational concepts.

Method

This research is a qualitative descriptive research. This study describes mathematics education students' mathematical computational thinking ability based on constructivist learning theory. The sample of this study amounted to 34 people who were taken purposively. The sample is selected purposively, and the researcher wants to obtain more in-depth information about the students' mathematical computational thinking ability to develop subsequent research.

This research was conducted at the Mathematics Education Study Program at Universitas Islam Riau, Pekanbaru, in the 2023/2024 academic year. There were 34 5th-semester students who took the Algebraic Structure course, and all were taken as samples for this study. Students were divided into 3 initial mathematical ability categories: low, medium and high. The value of students' initial mathematical ability is taken from the previous quiz score.

The mathematical computational thinking ability referred to in this study is, according to (Angeli et al., 2016; Csizmadia et al., 2015; Lee et al., 2022), computational thinking is a thought process that involves elements such as abstraction, generalisation, decomposition, algorithmic thinking, and debugging. According to Maharani et al. (2019), solving a problem through computational thinking can be seen from the following indicators: (1) Abstraction: Students can decide on an object to use or reject, which can be interpreted as separating important information from information that is not used; (2) Generalization: the ability to formulate solutions into a general form so that they can be applied to different problems, can be interpreted as the use of variables in solving solutions; (3) Decomposition: the ability to break down complex problems into simpler ones that are easier to understand and solve; (4) Algorithmic: the ability to design step by step an operation/action how the problem is solved; (5) Debugging: the ability to identify, discard, and correct errors.

The operational steps of implementing constructivism learning theory in research on the computational thinking ability of mathematics education students in the Algebraic Structure course can cover various aspects involving interactions between students, lecturers and subject matter. Here are some operational steps that can be applied (Cobb, 2014; Duffy & Jonassen, 2013; Josi & Patankar, 2016;

Piaget, 1976; Vygotsky, 1978): 1) learning materials designed to encourage knowledge construction; 2) Students can be encouraged to work together, discuss, and solve problems together with lecturer guidance; 3) Use open-ended questions and open-ended problems to stimulate critical thinking and drive understanding of mathematical and computational concepts; 4) Provide constructive feedback to students on their performance and how to improve their computational thinking skills.

The stages of learning in constructivism learning theory in this study are (Suprapti, 2018; Saputro & Pakpahan, 2021; Sari & Kurniawan, 2023): (1) Students connect with existing knowledge. They try to understand new concepts within their frame of reference; (2) Students actively seek information or experiences relevant to the concepts being studied. They conduct independent research, discussion, or exploration; (3) Students build their understanding through reflection, dialogue, or direct experience. They try to solve problems, connect information, and build new concepts; (4) Students critically reflect on their understanding. They dig deeper into concepts, refine existing understandings, or identify confusions; (5) Students apply the knowledge they construct in real situations or relevant problems. They seek to transfer their understanding to other contexts; (6) Students evaluate their understanding and performance in mastering the concepts or skills learned. They provide feedback to themselves or others. Furthermore, one example of a test instrument on computational thinking skills used in this study is as follows: 1) Show that G= {0,1}, (G, x) is not a group.

Results

This study aims to examine mathematics education students' mathematical computational thinking ability in the algebraic structure course, which is associated with constructivism learning theory. To see students' mathematical computational thinking ability on the material, a test related to mathematical computational thinking ability was given, the results of which are as follows:

Table 1. Description of Students' Mathematical Computational Thinking Ability

Descriptive statistics	Students
N	34
Mean	80.56
Sd	1.86
Max	100
Min	62

Based on the data in Table 1, it can be seen that, on average, students' mathematical computational thinking is already high; there are even students who reach a score of 100. Next, we will examine students' average mathematical computational thinking ability based on prior mathematical knowledge

(PMK). The prior mathematical knowledge is taken from the previous quiz score. The following is a description of students' mathematical computational thinking ability based on PMK:

Table 2: Average Mathematical Computational Thinking Ability of Students Based on PMK

Ability	N	Mean
High	11	93,09
Medium	12	79,67
Low	11	69

Based on the data in Table 2, the average mathematical computational thinking ability of students with high prior mathematical knowledge is much better than that of students with medium and even low prior mathematical knowledge. This shows that descriptively, PMK can also differentiate students' mathematical computational thinking. The students with a high level of ability in mathematical computational thinking, with an average score of about 93.09, showed a strong understanding of mathematical computational aspects. They have background knowledge and skills that support computational thinking. The group of students with moderate ability had an average of about 79.67. They have a good understanding of mathematical computing, but there is potential for further improvement and development. The low-ability group of students, with an average of around 69, showed a lower understanding of mathematical computation. This indicates a lack of knowledge or skills required in this context.

During the learning process in the Algebraic Structure course, lecturers conduct learning that emphasises Constructivism learning theory. Computational thinking is powerfully relevant to constructivist learning theory in education. Constructivism learning theory emphasises that students receive and actively construct knowledge through experience, reflection, and interaction. Here are some points of the relevance of computational thinking to constructivism learning theory (Vygotsky, 1978; Piaget, 1976; Krahenbuhl, 2016; Paradesa, 2015): (1) Constructivism learning theory emphasises student-centred learning, where students are actively involved in the learning process. Computational thinking allows students to actively engage in problem-solving, programming, and exploration of computational concepts, thus fitting this approach; (2) Constructivism theory states that students should construct their knowledge through reflection and understanding of concepts. Computational thinking involves knowledge construction where students create algorithms, programming, and problem-solving, all forms of knowledge construction; (3) Constructivism theory underlines the importance of students' active engagement in learning. Computational thinking encourages active engagement through programming, solution development, and technology exploration, supporting this theory; (4) Computational thinking provides students with practical experience. Constructivism learning theory emphasises the importance of hands-on experience in learning, and computational thinking provides practical problem-solving and programming experience.

Constructivism learning theory, first developed by Jean Piaget and later expanded by figures such as Lev Vygotsky, emphasises that learning is an active process in which individuals construct knowledge through experience and interaction with the environment. Individuals build understanding through reflection and the construction of knowledge centred on personal experience. The constructivist approach to mathematics learning emphasises the importance of deep understanding rather than factual memorisation. It recognises the critical role of practical experience, reasoning and collaboration in understanding mathematical concepts. Students are encouraged to construct their mathematical knowledge with the help of lecturers, peers and learning resources (Vygotsky, 1978; Piaget, 1976; Krahenbuhl, 2016; Cobb, 2014; Duffy & Jonassen, 2013; Josi & Patankar, 2016).

Computational thinking is a cognitive ability concerned with solving problems and designing solutions using concepts and techniques commonly associated with computing. It includes understanding the concept of algorithms, problem decomposition (breaking a problem into smaller parts), patterns, and abstraction (taking the core or essence of a problem). Computational thinking refers to a person's ability to think systematically, logically, and efficiently in solving problems, which can be applied in various contexts, including in programming, data science, and so on (Su & Yang, 2023; Ersozlu et al., 2023; Misirli & Komis, 2023). The following is the achievement of mathematical computational thinking ability in the Algebraic Structure course:

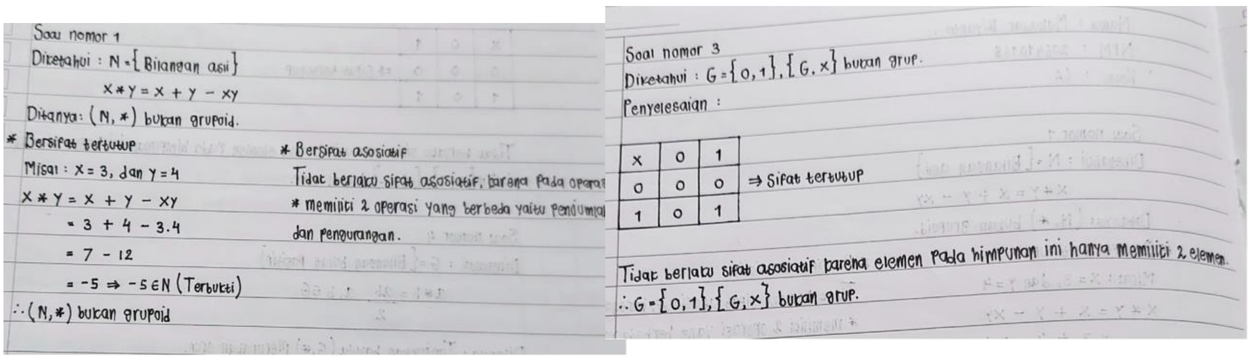

Figure 1. Students' Mathematical Computational Thinking Ability Test Answers

Based on the answers to the student's mathematical computational thinking ability test results above, it is obtained that the student can analyse mathematical problems or computational situations well. The student can decompose the problem into clear steps or algorithms to find a solution. The student may tend to think creatively when finding efficient and innovative solutions. The student can utilise existing knowledge and skills to face new challenges. In computational problem-solving, minor errors can lead to different results. Good test results signify an individual's ability to think carefully and accurately when executing computational algorithms and processes. Good computational thinking skills also reflect a strong understanding of the mathematical concepts underlying the computational method. The student can tackle complex mathematical or computational problems and solve them with strong thinking skills. Good test results may indicate that the student can execute computational processes quickly and efficiently.

Discussion

Differences in mathematical computational thinking ability between high, medium, and low-ability student groups can be influenced by several factors (Anistyasari et al., 2023; Saputro & Pakpahan, 2021; Sari & Kurniawan, 2022), including (1) Students with an educational background or prior knowledge in mathematical computing have an advantage. They have studied or have previous experience relevant to this topic; (2) Motivational factors in learning can affect computational thinking ability. Students who are highly motivated to understand this topic may be more likely to develop robust computational thinking skills; (3) Teaching methods and learning strategies used by lecturers or educational institutions can also play a role. A more interactive and practical approach to teaching mathematical computing might contribute to the development of better computational thinking skills; (4) Students who have access to relevant resources and tools for mathematical computing, such as software or libraries that support this topic, might be more likely to develop robust computational thinking skills; (5) Students' intrinsic ability to think abstractly, recognise patterns, and perform mathematical problem solving also plays a vital role in computational thinking skills; 6) The quality of instruction provided by lecturers or instructors can have a significant impact on students' understanding and development of computational thinking skills; (7) Collaboration and interaction between students in groups can influence the development of computational thinking skills. Learning through collaboration with peers with a high level of understanding might improve understanding; (8) Students' ability to organise and control their thought processes, known as metacognitive ability, is also essential in developing computational thinking skills.

Differences in computational thinking ability can be viewed from the constructivist learning theory, which emphasises that knowledge results from construction or formation. The following are some of the causes of differences in computational thinking ability in terms of constructivism learning theory (Sari & Kurniawan, 2023; Oktavianti, 2021; Umbara, 2017): (1) Constructivism theory emphasises that students must discover new things and develop understanding actively. In computational learning, students should be allowed to find new ways of solving problems and develop their computational skills; (2) Constructivism theory emphasises collaboration and active learning to improve computational thinking ability. In computational learning, students can work together to develop solutions and manage the problems faced; (3) Constructivism theory emphasises the process of discovery and social interaction to improve computational thinking ability. In computational learning, students can interact socially to develop ideas and solutions, which can be implemented in computational projects; (4) Opportunities for learners to interpret information. Constructivism theory emphasises that students can analyse data in their minds only in the context of their own experience and knowledge, needs, background and interests. In the context of computing learning, students should be given opportunities to interpret information and apply critical thinking.

Constructivism learning theory can help improve computational thinking ability by emphasising learner-centred learning, encouraging collaboration and active learning, discovery and social interaction, discovering new things, and assisting learners in actively building understanding. In computing learning, constructivism theory can help learners develop computational thinking ability through relevant experiences and activities and build their computational knowledge and skills. In addition, constructivism theory can also help learners generate innovative ideas and solutions to solve technological problems. Therefore, constructivist learning can help improve computational thinking ability by allowing learners to actively learn, discover new things, and collaborate with others to develop solutions and manage problems at hand.

Several critical points regarding the relationship between computational thinking and constructivist learning, as stated by (Amineh Davatgari, 2015; and Ackerman, 2001), are: (a) Constructivism highlights that optimal student learning occurs when they create tangible or meaningful objects. It stresses the importance of students actively engaging in creation, with others participating in the process, fostering genuinely meaningful learning. (b) Constructivist learning emphasises students' interaction with their artefacts and how this interaction promotes independent learning and the formation of new knowledge. (c) An extension of Piaget's constructivism, constructivism underscores that knowledge is actively generated by a child in their environmental context. (d) It also highlights the significance of tools, media, and conditions for human development in constructivist learning. (e) Constructivism is the foundation for constructionism, focusing on technology's role in aiding students' knowledge acquisition. Therefore, constructivist learning strongly correlates with computational thinking, offering a robust basis for its development in mathematics education. This aligns with Supriyadi and Dahlan's (2022) view that constructionism and constructivism significantly relate to computational thinking and mathematics education. This is supported by the increasing publication of diverse articles across various journals annually.

Ali and Yahaya’s study (2020) shows a notable correlation between computational thinking and constructivist learning. Constructivist learning underscores the idea that students actively construct their understanding through learning experiences and subsequent reflection on those experiences. When applied to computational thinking, the constructivist approach enables students to participate actively in problem-solving and programming and cultivate their computational thinking skills. Most experimental studies utilise constructivist learning theories, suggesting their effectiveness in fostering computational thinking. This implies that constructivist learning can establish a robust basis for nurturing computational thinking abilities, particularly among elementary and secondary school students. Therefore, Ali and Yahaya (2020) emphasise the significance of the constructivist approach in computational thinking development, highlighting that constructivist learning theories can serve as a potent foundation in curriculum design and teaching methodologies to facilitate the growth of computational thinking in students.

Voon et al. (2022) highlight a robust and mutually beneficial correlation between computational thinking and constructivist learning. The integration of computational thinking principles into constructivist learning involves a series of steps: identifying problems, breaking them down into manageable stages, formulating credible solutions, engaging in structured argumentation to present these solutions, assessing and appraising arguments from others, and ultimately refining these arguments and original solutions through abstraction and generalisation of concepts. The computational thinking framework allows students to rigorously investigate uncertainty by critically analysing data using evidence-based justification, navigating uncertainty, reconciling conflicting ideas with established claims, and making scientifically sound decisions based on evidence. This framework underscores the intricate connection between constructivist learning and computational thinking, demonstrating how students, through algorithmic thought, can evolve while crafting arguments in both spoken and written contexts.

Constructivism learning theory is a theory that gives freedom to humans who want to learn or seek their needs. This theory teaches that students should be actively involved in learning and build their understanding through interaction with prior knowledge. The following are things that should be considered in order to improve students' computational thinking skills in terms of constructivism learning theory (Yulianti & Sari, 2022; Amineh & Davatgari, 2015; Bada & Olusegun, 2015; Barac, 2017; Mattar, 2018): (1) Students should actively engage in computational learning and construct their own understanding through interaction with prior knowledge; (2) Computational learning should help students to develop higher order thinking skills in terms of critical thinking and problem solving; (3) Computational learning should teach computational thinking, i.e. how to think the way computer scientists think, to solve real world problems; (4) Computing learning should use a project-based learning approach, where students are given tasks to solve real problems using computational thinking; (5) Computing learning should use technology, such as software and hardware, to help students understand computing concepts; (6) Lecturers should provide constructive feedback and help students to improve their understanding of computational concepts.

The challenges in applying constructivism learning theory in learning computational thinking are as follows: (1) Constructivism learning requires longer time because students have to build their understanding through interaction with prior knowledge; (2) Lecturers who apply constructivism learning theory should be trained in managing learning and providing constructive feedback; (3) Computational learning requires adequate resources, such as software and hardware, to help students understand computational concepts; (4) Computational learning curriculum should be well developed and by constructivism learning theory. (5) Evaluation of computing learning should be done appropriately and per constructivism learning theory, i.e., by providing constructive feedback and helping students improve their understanding of computing concepts.

Here are some strategies that can be used to apply constructivism learning theory in learning computational thinking: (1) Computational learning should use a project-based learning approach, where students are given tasks to solve real problems using computational thinking; (2) Computational learning should teach computational thinking, which is a way of thinking the way computer scientists think, to solve real-world problems; (3) Computational learning should use technology, such as software and hardware, to help students understand computational concepts; (4) Lecturers should provide constructive feedback and help students to improve their understanding of computing concepts; (5) Computing learning should encourage students to think critically in solving problems and build their own understanding; (6) Lecturers should use discussion methods to help students build their own understanding of computing concepts; (7) Computing learning requires adequate resources, such as software and hardware, to help students understand computing concepts; (8) The computational learning curriculum should be well developed and in line with constructivism learning theory; (9) Students should be actively involved in learning and construct their own understanding through interaction with prior knowledge. By applying these strategies, learning computational thinking can be more effective and efficient in developing students' computational thinking skills. In addition, students will also be more skilful in critical thinking and problem-solving and able to teach computational thinking.

Conclusion

Based on the data processing results, it is obtained that the average mathematical computational thinking ability of students as a whole is good, and descriptively, the prior mathematical knowledge can also distinguish students' mathematical computational thinking. The group of students with a high level of mathematical computational thinking ability shows that they have a strong understanding of aspects of mathematical computing and background knowledge and skills that support computational thinking ability. The group of students with medium ability have a pretty good sense of mathematical computation, but there is potential for further improvement and development. The students in the low-ability group showed a lower knowledge of mathematical calculation.

Recommendations

Constructivism learning theory teaches that students should be actively involved in learning and construct their understanding through interaction with prior knowledge. In learning computational thinking, strategies such as project-based learning approach, teaching computational thinking, using technology, providing feedback, encouraging students to think critically, using discussion methods, using adequate resources, developing appropriate curriculum, and involving students actively can apply constructivism learning theory. However, there are some challenges in applying constructivist learning theory in learning computational thinking, such as requiring longer time, trained teachers, adequate resources, proper curriculum development, and proper evaluation.

References

Akkoç, H., & Ocak, G. (2019). Effect of Constructivist Learning Approach on Students' Academic Achievement in Mathematics. Journal of Education and Learning, 8(6), 60-72. DOI: 10.5296/jel.v8i6.15357

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 computational thinking curriculum framework: Implications for teacher knowledge. Educational Technology and Society, 19(3), 47–57. DOI: 10.1109/ICSE.2016.51

Amineh, J. R., & Davatgari, H. A. (2015). Review of Constructivism and Social Constructivism. Journal of Social Sciences, Literature and Languages, 1(1), 9-16.

Anistyasari, E., Ekohariadi, & Munoto. (2023). Strategi Pembelajaran untuk Meningkatkan Keterampilan Pemrograman dan Berpikir Komputasi: Sebuah Studi Literatur. Jurnal Pendidikan Teknik Elektro, 12(1), 37-44. DOI: 10.21831/jptk.v12i1.40260

Ackermann, E. (2001). Piaget's constructivism, Papert's constructionism: What is the difference—future of learning group publication, 5(3), 438.

Ali, W. N. W., & Yahaya, W. A. J. W. (2020). A systematic review of learning theory on computational thinking. Journal of Human Development and Communication, 9, 11-22.

Bada, S., & Olusegun, S. (2015). Constructivism Learning Theory: A Paradigm for Teaching and Learning. IOSR Journal of Research & Method in Education (IOSR-JRME), 5(6), 66-70.

Barr, V., & Stephenson, C. (2011). Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? ACM Inroads, 2(1), 48–54. https://doi.org/10.1145/1929887.1929905

Baran, A., & Maskan, A. (2021). The Effect of Constructivist Learning Approach on Students' Academic Achievement: A Meta-Analysis Study. Educational Psychology Review, 33(3), 803-824. DOI: 10.1007/s10648-021-09608-9.

Barac, M. (2017). Science Teacher Education in the Twenty-First Century: A Pedagogical Framework for Technology-Integrated Social Constructivism. Research in Science Education, 47, 283–303.

Brown, C., & Capper, J. (2019). Computational Thinking in Mathematics Education. ZDM Mathematics Education, 51(2), 181–193. DOI: 10.1007/s11858-018-0106-3

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). Computational Thinking: A Guide for Teachers. Computing At School, p. 18. DOI: 10.1002/9781118761706.ch2

Cobb, P. (2014). The tension between theories of learning and instruction in mathematics education. Educational Studies in Mathematics, 86(1), 81–95. DOI: 10.1007/s10649-013-9513-9.

Duffy, T. M., & Jonassen, D. H. (2013). Constructivism and the Technology of Instruction: A Conversation. Erlbaum, pp. 1–15.

Ersozlu, Z., Swartz, M., & Skourdoumbis, A. (2023). Developing Computational Thinking through Mathematics: An Evaluative Scientific Mapping. Education Sciences, 13(4), 1–15. DOI: https://doi.org/10.3390/educsci13040422

Grover, S., & Pea, R. (2013). Computational Thinking in K–12: A review of the state of the field. Educational Researcher, 42(1), 38–43. DOI: 10.3102/0013189X12463051

Josi, J. S., & Patankar, P. S. (2016). Use of Constructivist Pedagogy in Science Education. Aayushi International Interdisciplinary Research Journal (AIIRJ).

Kale, U., Akcaoglu, M., Cullen, T., Goh, D., Devine, L., Calvert, N., & Grise, K. (2018). Computational What? Relating Computational Thinking to Teaching. TechTrends, 62(6), 574–584. DOI: 10.1007/s11528-018-0290-9

Kania, N., & Kusumah, Y. S. (2023, November). Bibliometric analysis using R studio: Twenty-eight years of virtual reality research in math teaching. In *AIP Conference Proceedings* (Vol. 2909, No. 1). AIP Publishing.

Krahenbuhl, K. S. (2016). Student-centred Education and Constructivism: Challenges, Concerns, and Clarity for Teachers. The Clearing House: A Journal of Educational Strategies, 1-9.

Lee, I., & Yadav, A. (2016). Computational thinking in elementary and secondary teacher education. ACM Transactions on Computing Education (TOCE), 16(1), 5–12. DOI: 10.1145/2839509

Lee, J., Joswick, C., & Pole, K. (2022). Classroom Play and Activities to Support Computational Thinking Development in Early Childhood. Early Childhood Education Journal. DOI: https://doi.org/10.1007/s10643-022-01319-0

Listiasari, D., Nasrullah, A., Yendra, N., Rahmadani, K., & Egwuasi, P. I. (2023). Unlocking the Potential for Creative Thinking Ability and Self-Efficacy: Implementation of Numbered Head Together and Algebra Blocks in Middle School Students. International Journal of Mathematics and Mathematics Education, 1(3), 236–247. https://doi.org/10.56855/ijmme.v1i3.737

Liu, Y.-C., Huang, T.-H., & Sung, C.-L. (2021). The determinants of the impact of personal traits on computational thinking with programming instruction. Interactive Learning Environments, 29(8), 1–15. DOI: 10.1080/10494820.2021.1969475

Maharani, S., Kholid, M. N., NicoPradana, L., & Nusantara, T. (2019). Problem Solving in the Context of Computational Thinking. Infinity: Journal of Mathematics Education, 8(2), 109–116. DOI: 10.22460/infinity.v8i2.p109-116

Mattar, J. (2018). Constructivism and connectivism in education technology: Active, situated, authentic, experiential, and anchored learning. RIED. Revista Iberoamericana de Educación a Distancia, 21(2).

Misirli, A., & Komis, V. (2023). Computational thinking in early childhood education: The impact of programming a tangible robot on developing debugging knowledge. Early Childhood Research Quarterly, 65, 139–158. DOI: 10.1016/j.ecresq.2023.104214

Oktavianti, S. (2021). Pengaruh Model Pembelajaran Osborn dengan Teknik Mind Mapping terhadap Kemampuan Berpikir Kritis Matematis Mahasiswa Melalui Pendekatan Konstruktivisme. Jurnal Pendidikan Matematika dan Sains, 6(1), 1-10. DOI: 10.24252/jams.v6i1.15508

Paradesa, R. (2015). Kemampuan berpikir kritis matematis mahasiswa melalui pendekatan konstruktivisme pada matakuliah matematika keuangan. Jurnal Pendidikan Matematika RAFA, 1(2), 306-325.

Piaget, J. (1976). Piaget's theory. In Handbook of Child Psychology, 1, 703–732. DOI: 10.1037/11361-000

Sardin, S., Dewi, S. P., Saleh, M., & Alrahhal, M. (2023). The Guided Note-Taking Learning Model Effectively Improves Students' Mathematics Learning Creativity. International Journal of Mathematics and Mathematics Education, 1(3), 236–249. https://doi.org/10.56855/ijmme.v1i3.738

Sari, D. P., & Kurniawan, A. (2022). Pengembangan Kemampuan Berpikir Komputasi Siswa Melalui Pembelajaran Berbasis Proyek. Jurnal Pendidikan Informatika dan Sains, 11(1), 1-10. DOI: 10.24815/jp.v11i1.20222

Sari, D. P., & Kurniawan, A. (2023). Penerapan Teori Belajar Konstruktivisme dalam Pembelajaran Berpikir Komputasi. Jurnal Pendidikan Informatika dan Sains, 12(1), 1-10. DOI: 10.24815/jp.v12i1.27016

Saputro, M. N. A., & Pakpahan, P. (2021). Pembelajaran Berbasis Proyek untuk Meningkatkan Kemampuan Berpikir Komputasi Siswa. Jurnal Pendidikan Informatika dan Sains, 10(2), 1-10. DOI: 10.24815/jp.v10i2.20219

Selden, A., Selden, J., & Mason, A. (2014). A programming-based approach for teaching mathematics to mathematics education students. International Journal of Mathematical Education in Science and Technology, 45(6), 890-900. DOI: 10.1080/0020739X.2014.915701

Su, J., & Yang, W. (2023). A systematic review of integrating computational thinking in early childhood education. Computers and Education Open, 4, 100122. DOI: 10.1016/j.caeo.2023.100122

Suprapti, E. (2018). Penerapan Teori Belajar Konstruktivisme untuk Meningkatkan Hasil Belajar Mahasiswa pada Mata Kuliah Matematika Dasar. Jurnal Pendidikan Matematika dan Sains, 3(1), 1-10. DOI: 10.24252/jams.v3i1.5853

Supriyadi, E., & Dahlan, J. A. (2022). Constructionism and Constructivism in Computational Thinking and Mathematics Education: Bibliometric Review. Jurnal Matematika dan Pendidikan Matematika, 12(1). https://doi.org/10.20961/jmme.v12i1.61946

Umbara, U. (2017). Implikasi Teori Belajar Konstruktivisme dalam Pembelajaran Matematika. Jumlahku Jurnal Matematika Ilmiah STKIP Muhammadiyah Kuningan, 3(1). DOI: 10.30598/jumlahku.v3i1.116

Voogt, J., Fisser, P., Tondeur, J., & Braak, J. V. (2015). Exploring the factors influencing teachers' innovative behaviour in technology-rich classrooms. Computers & Education, 91, 1-12. DOI: 10.1016/j.compedu.2015.09.001

Voon et al. (2022). Developing Computational Thinking Competencies through Constructivist Argumentation Learning: A Problem-Solving Perspective. International Journal of Information and Education Technology, 12(6), 529–539.

Vygotsky, L. S. (1978). Mind in society: The development of higher psychological processes. Harvard University Press.

Yılmaz, R. (2017). Constructivist learning theory and the design of digital math games: Mathematics education at play. European Journal of Science and Mathematics Education, 5(4), 341–351. DOI: 10.30935/sci-math/9624

Ye, H., Liang, B., Ng, O.-L., & Chai, C. S. (2023). Integration of computational thinking in K-12 mathematics education: a systematic review on CT-based mathematics instruction and student learning. International Journal of STEM Education, 10(1), 1–26. https://doi.org/10.1186/s40594-023-00396-w

Yulianti, D., & Sari, D. P. (2022). Penerapan Model Pembelajaran Berbasis Masalah untuk Meningkatkan Kemampuan Berpikir Komputasi Siswa. Jurnal Pendidikan Informatika dan Sains, 11(2), 1-10. DOI: 10.24815/jp.v11i2.20228

Yustitia, et.al. (2024). *Pendidakan di Era Digital.* EDUPEDIA Publisher

Author Information

Lilis Marina Angraini
https://orcid.org/0000-0003-1328-5466
Universitas Islam Riau
Jalan Kaharuddin Nasution Nomor 113
Pekanbaru Riau
Indonesia
Contact e-mail: *lilismarina@edu.uir.ac.id*

Nia Kania
https://orcid.org/0000-0002-1994-6635
Universitas Majalengka
Jl. K. H. Abdul Halim No. 103 Majaelngka, Jawa Barat
Indonesia
Contact e-mail: *niakania@unma.ac.id*

Ferit Gürbüz
https://orcid.org/0000-0003-3049-688X
Department of Mathematics, Kırklareli University, Kırklareli 39100,
Türkiye
Contact e-mail: *feritgurbuz@klu.edu.tr*