p-ISSN: 2086-4280
e-ISSN: 2527-8827

Students' Computational Thinking Ability in Calculating an Area Using The Limit of Riemann Sum Approach

Enjun Junaeti[1*], Tatang Herman[2], Nanang Priatna[3], Dadan Dasari[4], Dadang Juandi[5]

[1*,2,3,4,5]Mathematics Education, Universitas Pendidikan Indonesia
Jalan Dr. Setiabudi No. 229, Bandung, Jawa Barat, Indonesia
[1*]enjun@upi.edu; [2]tatangherman@upi.edu; [3]nanang_priatna@upi.edu; [4]dadan.dasari@upi.edu; [5]dadang.juandi@upi.edu

Article received: 2022-12-22, revised: 2023-04-20, published: 2023-04-30

Abstrak

Melatih kemampuan berpikir komputasional mahasiswa membuka peluang untuk lebih menguasi konsep, menganalisis permasalahan, dan membangun solusi dunia nyata. Tujuan penelitian adalah menganalisis kemampuan berpikir komputational mahasiswa Pendidikan Ilmu Komputer berupa kemampuan abstraksi, dekomposisi, berpikir algoritmik, dan generalisasi. Metode penelitian yaitu studi kasus dengan pendekatan kualitatif deskriptif. Pembelajaran dilakukan kepada 40 mahasiswa semester 1 (satu) secara kolaboratif dalam penyelesaian masalah luas daerah dengan pendekatan limit. Pada akhir pembelajaran mahasiswa diberikan soal tes kemampuan berpikir komputasional mahasiswa. Jawaban tes setiap mahasiswa dianalisis dari segi sisi fungsi mental yang muncul untuk mengetahui karakteristik akusisi kemampuan penyelesaian masalah. Pada penelitian yang telah dilakukan mahasiswa dikategorikan dalam kelompok novice, advanced beginner, competent, proficient, dan expert berdasarkan karakter penyelesaian masalahnya. Pada umumnya setiap mahasiswa telah memiliki kemampuan berpikir algoritmik. Sebagian besar mahasiswa (kecuali kategori novice) juga telah mampu mengabstraksi dan mendekomposisi permasalahan. Sedangkan kemampuan pengenalan pola baru terlihat pada mahasiswa dengan kategori competent, proficient, dan expert.
Kata Kunci: Berpikir komputasional; Jumlah Riemann; Luas daerah; Problem solving.

Abstract

Training students' computational thinking ability provides opportunities to comprehend concepts, analyse problems, and build solutions in real-life contexts. The purpose of the study was to analyse the computational thinking abilities of Computer Science Education students, i.e., abstraction, decomposition, algorithmic thinking, and generalization abilities. The research method used was a case study with a descriptive qualitative approach. The learning process was conducted by 40 students for semester 1 (one) semester collaboratively in solving area problems using the limit approach. At the end of the lesson, the students were tested through students' computational thinking abilities. Each student's answers were analyzed in terms of the mental functions that emerged to determine the characteristics of the acquisition of problem-solving ability. In this study, the students were categorized into groups of novices, advanced beginner, competent, professional, and expert based on the natures of their problem solving. In general, every student had the ability to think algorithmically. Most students (except the novice category) were able to abstract and unravel the problems. Meanwhile, the ability to recognize new patterns were demonstrated by the students in the competent, professional, and expert categories.
Keywords: Area; Computational thinking; Problem solving; Riemann Sum.

I. Introduction

Computational thinking (CT) is a thought process that involves problem formulation and its solution interpretation as a transformation of the information that can be effectively performed by the agents (Wing, 2006; Rahayu, Liddini, & Maarif, 2022). Wing also revealed that CT is an analytical ability that every child should have in addition to reading, writing, and arithmetic. Educational researchers and practitioners advocate the introduction of CT in education to foster problem solving and creativity among learners (Huang et al, 2022; Israel-Fishelson et al, 2020). The goal of developing CT is to assist students to understand the basic principles of how computers process information and use this knowledge to solve problems in everyday life (Wing, 2011; Gustiani & Puspitasari, 2021). Training students' CT skills provides opportunities for students to fully understand the concepts, analyze problems, and build solutions in the real world (Seehorn et al., 2011). Through CT, the students are encouraged to develop important skills such as logical reasoning, analysis, and creativity, which benefits various fields and situations.

The definition of CT in the learning process is evolving along with its implementation in the educational research. Various definitions as well as frameworks for operational definitions of CT have been proposed in several related literatures (Barr & Stephenson, 2011; Csizmadia et al, 2015; Marques et al, 2018; C. C. Selby, 2015). However, the most common description and definition of CT are associated with abstraction, algorithmic thinking, decomposition, and pattern recognition (Boom et al., 2018). (Boom et al., 2018). An illustration of CT capabilities is represented in Figure 1.

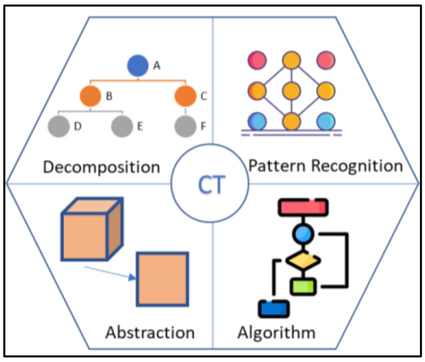

Figure 1. CT Ability Illustration.

Cetin and Dubinsky (2017) define abstraction in CT with the terms *extraction*, *decontextualization,* and *essence*. In the implementation of Boom et al. (2018) and Csizmadia et al. (2015) interpret the ability to think in abstraction as an ability to choose a good representation. Selby and Woollard (2013) defined the ability to decompose problems as the ability to break down problems into simpler components. Csizmadia et al. (2015) described algorithmic thinking as the ability to define steps clearly in order to get a problem solution. Meanwhile, pattern recognition consists of the ability to identify, generalize and utilize patterns.

Computational thinking is considered to be a valuable tool for students in mathematics learning due to the fact that it fosters important skills such as logical reasoning, analysis, and problem solving (Eisenberg, 2002; Hadjerrouit & Hansen, 2022; Lockwood et al, 2016; Lu et al, 2022; Sung & Black, 2021). These skills can be applied to a variety of mathematical concepts and to a greater extent help students understand and solve mathematical problems. By integrating

p-ISSN: 2086-4280
e-ISSN: 2527-8827

computational thinking into mathematics education, students can develop a deeper understanding of mathematics and be better equipped to apply these skills in real-world contexts.

Meanwhile, mathematics can be a meaningful tool to foster computational thinking skills considering it trains students to formulate problems and search for solutions in a structured and meaningful way (Benakli et al, 2017; Gadanidis, 2017; Pei et al, 2018; Rambally, 2017; Rodríguez del Rey et al, 2021; Pipitgool et al, 2021; Son & Lee, 2016). By doing math problems, students learn how to approach problems logically and systematically, and apply these skills to a variety of real-world situations (Afriansyah & Turmudi, 2022). By integrating computational thinking into mathematics education, students can develop a deeper understanding of mathematics and problem solving, and be better equipped to apply these skills in studies and careers in the future.

This mutual relationship between CT (computational thinking) and mathematics is the rationale of this study to investigate the two variables. By introducing problem solving as a teaching method, students are encouraged to build CT skills while learning Mathematics. This study aimed to integrate CT and Mathematics learning explicitly to show that students ar able to develop CT and Mathematics concepts comprehensively. The purpose of this study was to analyze the potential CT skills of Computer Science Education students demonstrated through the ability to analyze problems, recognize and generalize patterns, think algorithmically, and abstract in problem solving based on the ability to acquire a skill.

II. METHODS

The study used a case study (Fraenkel, Wallen, & Hyun, 2012), in which the researcher analyzed the answers regarding the area calculation using the Rieman sum approach formulated by students in depth and detail. The researchers then analyzed the data with a qualitative approach to identify the level of problem-solving ability and the construction of CT ability demonstrated in the solution of the given problem. The participants consisted of 42 Computer Science Education students in Bandung city who were taking Calculus in semester 1 (one). The students consisted of 19 males and 23 females with an age range of 17 to 19 years. One of the researchers acted as a facilitator (lecturer) in classroom during the lessons.

Learning activities began with group assignments consisting of 4 (four) to 5 (five) people. This assignment was designed to present a sequential problem solving as a stage of problem-solving construction with the idea of scaffolding or Vygotsky's idea of mediation (Verenikina, 2008). The assignment was given a week before the learning activity. The flow of scaffolding to find the area with the limit approach is provided in Figure 2. Scaffolding stage 1 was conducted to assist the students to recognize the area to be calculated. Stage 2 was a strategy for solving area problems by dividing the problem on a smaller scale, while stage 3 was to obtain data that would be used based on the strategy in stage 2.

Scaffolding stage 4 was conducted to obtain the solution.

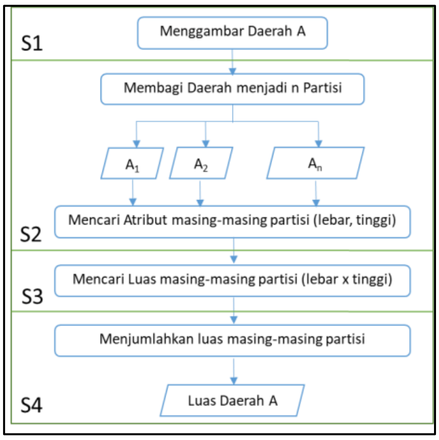

Figure 2. Scaffolding Stage to Calculate an Area.

During the lessons, the students were facilitated to have a discussion regarding the solved problem and make the generalization of the necessary concepts, including the formation of partitions, the shape of the approach area, and the limit of the total area of each partition when the number of partitions approaches infinity. There were no other researchers present during the lessons, but the researchers recorded the learning activities in the form of a voice recording with the permission of the participants.

After the learning activities, students were given a test to do in 30 minutes. The test was conducted to identify the ability to solve problems of the area provided by using the limit sum approach. The test questions given are presented in Figure 3.

Hitunglah luas daerah yang dibatasi oleh kurva $f(x) = x^2-4$, sumbu-x, dan sumbu-y, pada interval [0,2] dengan membagi daerah tersebut menjadi:

a. 4 partisi
b. 100 partisi

Figure 3. Test Items

Based on the test, the researchers categorized the answers based on the level of problem solving ability adjusted to the ability acquisition model developed by Dreyfus and Dreyfus (2004; 1980). This model concerned with four mental functions, namely Novice, Advanced Beginner; Competent, Proficient, and Expert. This model focused on four mental functions, namely:

1. *recollection* (the process of retrieving previously learned information from memory)
2. *recognition* (the ability to identify information or stimuli that have been encountered before),
3. *decision* (involves choosing between different options or actions) and
4. *awareness* (refers to the state of being aware or paying attention to one's surroundings, thoughts, and emotions and how they vary at each skill level) (Honken, 2013).

With each increase of the skill level, one of the mental functions matures. Table 1 illustrates the changes in expertise of the mental functions at each stage.

Table 1.
The characteristics of problem-solving ability

Kategori	Fungsi Mental			
	Recl	Recg	Dec	Aw
Novice	NS	Dc	An	Mt
Advanced Beginner	S	Dc	An	Mt
Competent	S	Hl	An	Mt
Proficient	S	Hl	In	Mt
Expert	S	Hl	In	Ab

Recl: *Recollection*; Recg: *Recognition*;
Dec: *Decision*; Aw: *Awareness*
NS: *Non-Situational*; S: *Situational*;
Dc: *Decomposed*; Hl: *Holistic*; An: *Analytical*;
In: *Intuitive*; Mt: *Monitoring*; Ab: *Absorbed*

Suppose in someone who is in the *Novice* category, then someone's mental recollection function is in a non-situational

p-ISSN: 2086-4280
e-ISSN: 2527-8827

condition, while the mental recognition function is in a *decomposed* condition. In the mental *decision* function, someone who is in the *novice* category solves problems analytically and his awareness is still in the monitoring condition. As a guide in categorizing the answers (problem solving), the characteristics of each category of solving the problem of area with the limit number approach are defined in Table 2. The definition of characteristics referred to a combination of assessment rubrics from the ability acquisition model that had been developed (Dreyfus & Dreyfus, 2005; Honken, 2013; Rousse & Dreyfus, 2021).

Table 2.
Characteristics of Area Problem Solving Ability

Kategori	Karakteristik
Novice	Deciphered the area problem solving on *context-free features*, without looking at the area to be found, and only based on the known formula.
Advanced Beginner	Recognize the area to be found (not *context-free*), but not yet be able to connect the partition area with sigma notation.
Competent	Identify the area to be found, b be able to connect the partition area with sigma notation (*holistic*), but the strategy is still procedural.
Proficient	Recognize the area to be found and be able to relate the partition area to sigma notation (holistic). His repertoire of situations is so extensive that usually any given situation immediately determines the appropriate action intuitively
Expert	No longer needing principles, stop paying conscious attention to his performance and let all the mental energy previously used to monitor the performance result almost instantaneously in the right perspective and its associated actions.

In the final stage, an analysis was conducted to explore the construction of students' CT abilities. Indicators of computational thinking ability adapted from Csizmadia, et al. (2015) in solving extensive problems are presented in Table 3 which is an adaptation and description of CT abilities.

Table 3.
The Indicators of Computational Thinking Ability

Sc	Indicators	CT
1	(1) describing the boundary of area, A (2) Marking area, A	A
2	(1) Dividing the area into k (k=4 or k=100) partitions (2) Dividing the area into n partitions (3) Finding the width of the partitions (k=4, 100) (4) Appling the partitions width formula (k=n) (5) Finding the partition boundary points (6) Obtaining the formula of partition boundary points (7) Finding the function values of the partition's boundary points (8) Acquiring function value patterns from partition boundary points	D, At D, P At P At P At P
3	(1) Calculating the area of each partition (2) Acquiring the area pattern of a partition	At P
4	(1) Sum the area of each partition to attain the area (2) Use sigma notation and its properties to sum the area of each partition	At A, D, At, P

Sc:Scaffolding; A: Abstraksi; D: Dekomposisi;
At: Algorithmic thinking; P: pattern recognition;

III. RESULT AND DISCUSSION

Based on the test conducted at the end of the lessons, the researcher categorized the answers based on the level of problem-solving ability adopted from ability acquisition model. Furthermore, from each category of answers, the potential of computational thinking ability demonstrated was identified. The results of the problem solution identification as well as the potential of students' computational thinking skills (abstraction, decomposition, algorithmic thinking, and pattern recognition) are as follows.

A. Problem-solving ability

Based on the answers given by the students, the researchers categorized the level of problem solving ability as follows.

1. *Novice*

The novice category was given to the students with the achievement of being able to find the data (width and height of the partition) required by using the rules but not understanding the problem based on the context of the given area (*free context fiture*). In this category there were 7 (seven) students. One of the answers of novice students is presented in Figure 4.

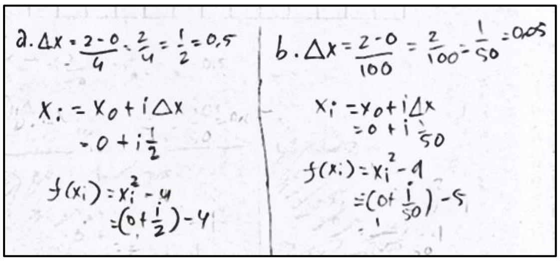

Figure 4. R1’s Answer Sheet in The Novice Category.

On the answer sheet, the students were able to find the partition boundaries and function values. However, they did not carefully analyze the area to be solved, they were unable to find the area to be, either by partition or as a whole. This was in line with the idea of (Honken, 2013) that at the *novice* level, an individual could only solve a small part of the problem without looking at the problem as a whole.

2. *Advanced Beginner*

The advanced beginner category was given to nine (9) students with the following characteristics: understanding the context of the area (not context free) to be found; recognizing the patterns of the procedures performed, but still limited to specific things. Examples of the answer sheets for advanced beginner students is presented in Figures 5 and 6.

The advanced beginner student group was able to find the area which the number of partitions was still small (easily summed without using the basic concept of sigma notation). The reason behind this might lead to the condition in which even though the students had been able to recognize the area pattern of each partition, they were unable to associate the summation of the partition areas with the sigma notation.

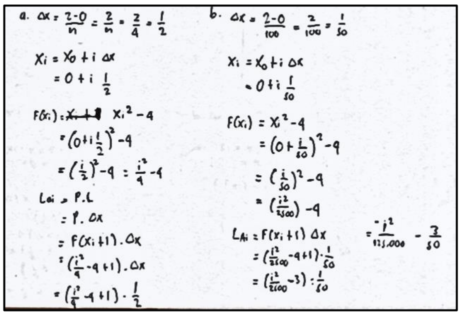

Figure 5. R2’s Answer Sheets in The *Advanced Beginner* Category.

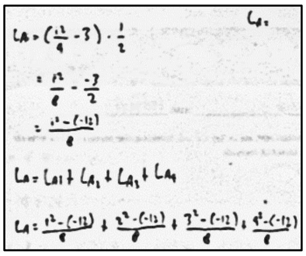

Figure 6. R2’s Answer Sheets in The *Advanced Beginner* (Continued).

According to Rousse and Dreyfus (2021), during the lessons, the lecturers should act as instructors assisting students to select and recognize relevant aspects as the organizer and the first source of the material, and a facilitator who provided many case examples for this level.

p-ISSN: 2086-4280
e-ISSN: 2527-8827

3. Competent

The students in the competent category with a total of 6 (six) people demonstrated these characteristics: understand the context of the area to be sought; recognize the patterns of the procedures performed and are able to connect the sum of the area of each partition with sigma notation. The examples of the answer sheets for competent students are presented in Figures 7 and 8.

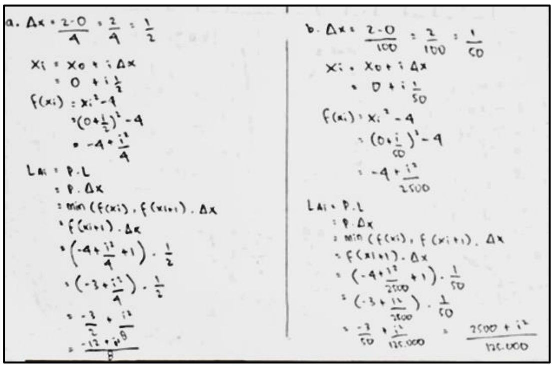

Figure 7. R3's Answer Sheet in The Competent Category.

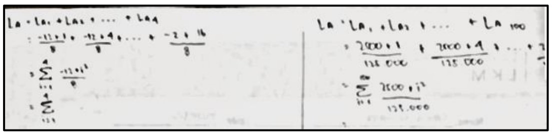

Figure 8. R3's Answer Sheet in The Competent Category (Continued).

The competent student group was able to find the area which the number of partitions was small or large by using the concept of sigma notation. However, in solving the problem, the students in this category were not yet able to use problem generalization strategies. So that problem solving is only done procedurally.

This was in line with the ability acquisition model in Table 2 adapted from Honken (2013). The model illustrated that the students in the competent category could determine the right action to deal with different situations, meaning that they were able to solve problems in accordance with the context of the problem. Referring to Table 1 regarding mental functions, Dreyfus (2004) stated that students who were in the *competent* category had not been able to make intuitive decisions, so that in solving the problem, it was solved *procedurally* without using generalizations from the patterns they had found.

4. Proficient

In the proficient category, there were 9 (nine) students with characteristics: understand the context of the area to be sought; recognize patterns from the procedures performed; able to connect the sum of the area of each partition with sigma notation and able to use generalization strategies for cases with many partitions. The examples of the answer sheets for proficient students are presented in Figures 9 and 10.

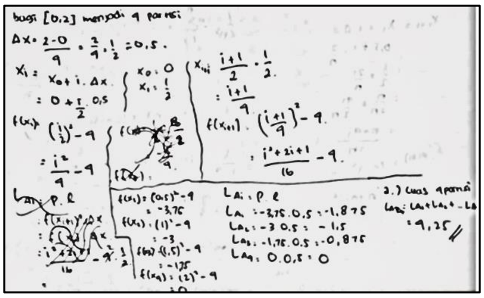

Figure 9. R4's Answer Sheets in Proficient Category.

The students in the proficient category were able to use generalization strategies intuitively from problems with a large number of partitions. However, this intuition had not yet become a spontaneous action in problem solving, so that for problems with a small number of partitions it was still proceeded procedurally (Rousse & Dreyfus, 2021).

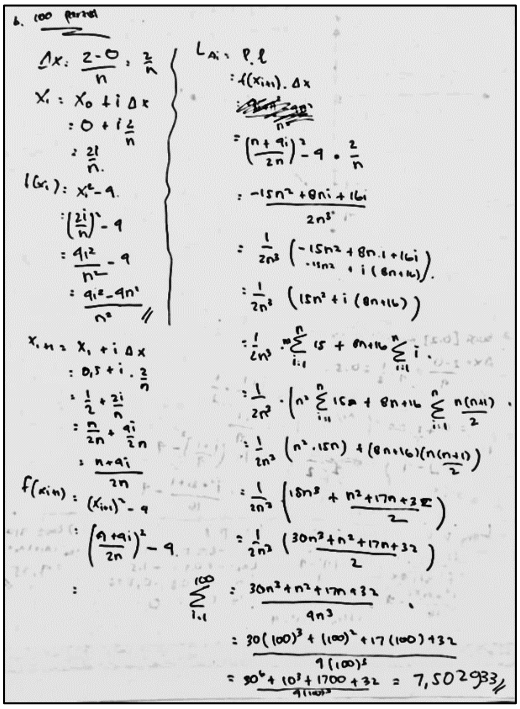

Figure 10. R4's Answer Sheets in Proficient Category (Continued).

5. *Expert*

The expert category was given to students with characteristics: understand the context of the area to be sought; recognize the patterns of the procedures performed; able to connect the sum of the area of each partition with sigma notation and able to use the instinct of generalization strategies for both small and large number of partitions. So that in solving the problem, students first solved the problem in general (n partitions) and then calculated the specific partition (*absorbed*). The examples of answer sheets for expert group students are presented in Figures 11 and 12.

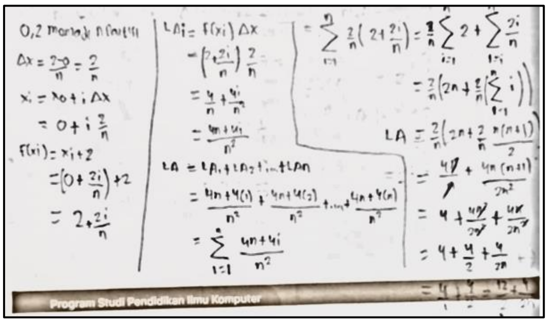

Figure 11. R5's Answer Sheets in Expert Category.

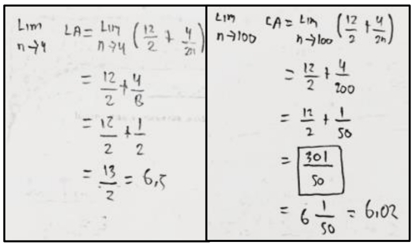

Figure 12. R5's Answer Sheets in Expert Category (Continued).

Of the 41 students, there were 2 (two) students whom the researchers could not identify their problem solving abilities. This occurred because the students returned the test answer sheet without a solution. The researchers were unable to confirm to the students for some reasons. This was supported by the testimony given by fellow students in his class that other lecturers also had difficulty communicating with these students.

B. Computational thinking ability

For each category of problem solving, the researchers constructed the CT flow of the procedure performed. Based on the construction of these flows, then analyzed the potential CT abilities owned by each category.

1. *Novice*

The construction of the flow of thinking in novice category is presented in Figure 13.

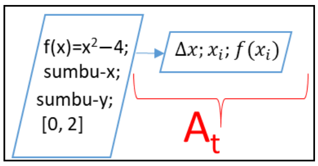

Figure 13. The Construction Flow of The Student's' Thinking in The Novice Category.

The students in the novice category demonstrated the potential ability to think algorithmically (At). The ability to decompose problems was not present, because the stages of problem solving were only conducted procedurally without analyzing the context of the problem (the broad search area). Likewise, the ability of abstraction and pattern recognition were not demonstrated by the students in this category. Thus, the students were only able to think algorithmically based on the problem solving process but did not analyze the context of the problem. (Rousse & Dreyfus, 2021).

2. Advanced Beginner

In the advanced beginner category, the students were able to recognize the context of the problem (Dreyfus & Dreyfus, 2005; Honken, 2013). This implied that the students were able to abstract (A) the problem to be solved. In addition, referring to Table 3 and the problem solution, students demonstrated the potential ability to think algorithmically (At) by solving the problem procedurally. Decomposition ability (D) was present when the students were observing the attribute value of each partition. However, there was no pattern recognition in the answers given. The construction of the flow of thinking in *advanced beginner* category students is illustrated in Figure 14.

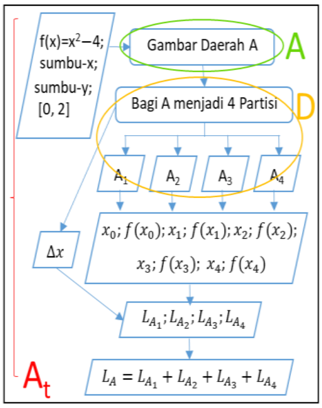

Figure 14. The Students' Construction Flow in The Advanced Beginner Category.

3. Competent

The flow of the students' thinking in the competent category is illustrated in Figure 15. It involved the four potential CT abilities (A, D, P, At). The ability to think algorithmically was demonstrated from the problem-solving procedure performed. The problem context had been extracted well. Decomposition and pattern recognition skills were demonstrated while the students were observing the attribute value of each partition, as well as the stages in calculating the total area of each partition (algorithmic) and were able to associate it with the sigma notation (abstraction). Despite in the problem solving, this abstraction was not included in the problem-solving strategy.

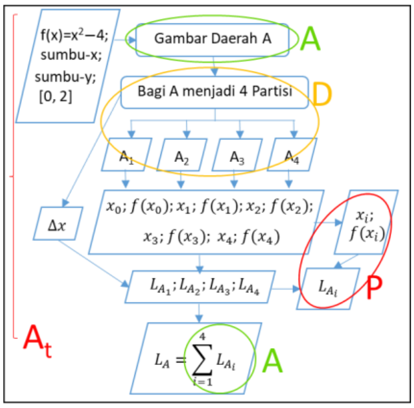

Figure 15. The Students' Construction Flow in The Competent Category.

4. *Proficient*

Like the students in the competent category, the four CT competencies (A, D, P, At) were already owned by the students in the proficient category, but they were still fixated on procedures. Nevertheless, the abstraction thinking process (A) that had been carried out by the students in the proficient category was implemented in the problem solving startegy, although only in the partition category with a large number of partitions.

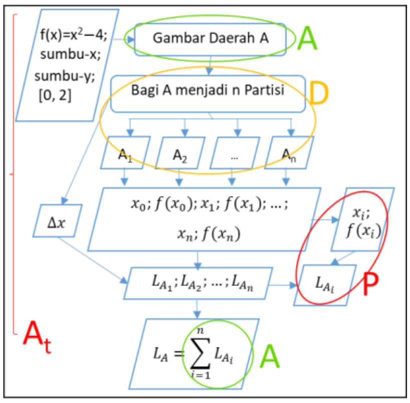

Figure 16. The Students' Construction Flow in The Proficient Category.

5. *Expert*

Among the students of the expert category, all four (A, D, P, At) CT abilities were demonstrated and even used to solve problems appropriately. The construction of the flow of thinking in the expert category is illustrated in Figure 17.

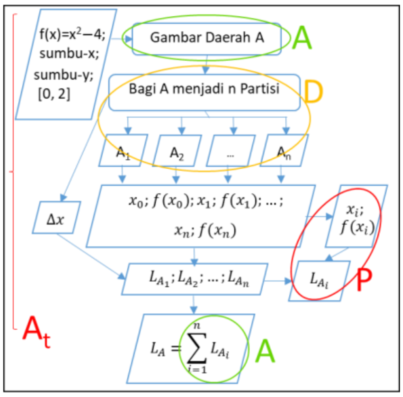

Figure 17. The Students' Construction Flow in The Expert Category.

According to the findings of the study, there were 5 (five) variations of problem solving with the categories of *novice*, *advanced beginner*, *competent*, *proficient*, and *expert*. For students in the *novice* and *advanced beginner* categories, they were unable to solve the problem completely. While in the *competent*, *proficient*, and *expert* categories, they solved completely, even in the *expert* category, demonstrated appropriate strategies on the problem-solving process.

As stated in Gadanidis (2017) abstraction ability is the main key in representing knowledge, this was in line with the findings found among the students in the novice category being unable to recognize the problems to be solved. The ability to recognize new patterns was demonstrated among the students in *competent*, *proficient*, and *expert* categories due to the fact that their mental recognition function had been in a holistic situation (able to see the problem as a whole) not just per part (Rousse & Dreyfus, 2021). The characteristics of the computer students who had been accustomed to an organized way of working and dividing problems into smaller parts in programming (Caeli and Yadav, 2020) were in line with the findings that each category of students already demonstrated the potential for decomposition and algorithmic thinking ability.

IV. CONCLUSION

The computational thinking ability demonstrated among the participants were algorithmic thinking ability. Most students (except the novice category) performed the decomposition ability. Likewise with the ability of abstraction, although in the advanced beginner category students were only able to abstract the problem. While the ability of pattern recognition was observed among the students in the competent, proficient, and expert categories.

The limitation of this study lies in the limited time of the test implementation and conducted after direct learning. So, it is possible that the understanding possessed

p-ISSN: 2086-4280
e-ISSN: 2527-8827

by students was a temporary state not an actual understanding even though there was no influence on the construction of thinking. Further research is recommended to also measure understanding and analyze student learning obstacles, especially the obstacles to the thinking process that cannot be achieved.

REFERENCES

Afriansyah, E. A., & Turmudi, T. (2022). Prospective teachers' thinking through realistic mathematics education based emergent modeling in fractions. *Jurnal Elemen*, *8*(2), 605-618.

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

Benakli, N., Kostadinov, B., Satyanarayana, A., & Singh, S. (2017). Introducing computational thinking through hands-on projects using R with applications to calculus, probability and data analysis. *International Journal of Mathematical Education in Science and Technology*, *48*(3), 393–427. https://doi.org/10.1080/0020739X.2016.1254296

Boom, K.-D., Bower, M., Arguel, A., Siemon, J., & Scholkmann, A. (2018). Relationship between Computational Thinking and a Measure of Intelligence as a General Problem-Solving Ability. *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education*, 206–211. https://doi.org/10.1145/3197091.3197104

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). *Computational thinking - a guide for teachers*. Computing at School. https://eprints.soton.ac.uk/424545/

Dreyfus, H. L., & Dreyfus, S. E. (2004). The Ethical Implications of the Five-Stage Skill-Acquisition Model. *Bulletin of Science, Technology & Society*, *24*(3), 251–264. https://doi.org/10.1177/0270467604265023

Dreyfus, H. L., & Dreyfus, S. E. (2005). Peripheral Vision: Expertise in Real World Contexts. *Organization Studies*, *26*(5), 779–792. https://doi.org/10.1177/0170840605053102

Dreyfuss, S. E., & Dreyfus, H. L. (1980). A five-stage model of the mental activities involved in directed skill acquisition. In *Operations Research Center* (Issue February).

Eisenberg, M. (2002). Output Devices, Computation, and the Future of Mathematical Crafts. *International Journal of Computers for Mathematical Learning*, *7*(1), 1–44. https://doi.org/10.1023/A:1016095229377

Gadanidis, G. (2017). Artificial intelligence, computational thinking, and mathematics education. *The International Journal of Information* *and Learning Technology*, *34*(2), 133–139. https://doi.org/10.1108/IJILT-09-2016-0048

Gustiani, D. D., & Puspitasari, N. (2021). Kesalahan Siswa dalam Menyelesaikan Soal Matematika Materi Operasi Pecahan Kelas VII di Desa Karangsari. *Plusminus: Jurnal Pendidikan Matematika*, *1*(3), 435-444.

Hadjerrouit, S., & Hansen, N.-K. (2022). *Undergraduate Mathematics Students Engaging in Problem-Solving Through Computational Thinking and Programming: A Case Study BT - Orchestration of Learning Environments in the Digital World* (D. Ifenthaler, P. Isaías, & D. G. Sampson (eds.); pp. 197–214). Springer International Publishing. https://doi.org/10.1007/978-3-030-90944-4_11

Honken, N. (2013). Dreyfus Five-Stage Model of Adult Skills Acquisition Applied to Engineering Lifelong Learning. *ASEE Conferences*. https://doi.org/10.18260/1-2--19457

Huang, M., Fan, R., Park, E., & Cheon, J. (2022). Exploring Instructional Strategies for Computational Thinking Concepts and Practices in Higher Education. In E. Langran (Ed.), *Proceedings of Society for Information Technology & Teacher Education International Conference 2022* (pp. 27–32). *Association for the Advancement of Computing in Education (AACE)*. https://www.learntechlib.org/p/220704

Israel-Fishelson, R., Hershkovitz, A., Eguíluz, A., Garaizar, P., & Guenaga, M. (2020). The Associations Between Computational Thinking and Creativity: The Role of Personal Characteristics. *Journal of Educational Computing Research*, *58*(8), 1415–1447. https://doi.org/10.1177/0735633120940954

Lockwood, E., Asay, A., DeJarnette, A. F., & Thomas, M. (2016). Algorithmic thinking: An initial characterization of computational thinking in mathematics. *38th Annual Meeting of the North American Chapter of the International Group for the Psychology of Mathematics Education*, 1588–1595.

Lu, C., Macdonald, R., Odell, B., Kokhan, V., Demmans Epp, C., & Cutumisu, M. (2022). A scoping review of computational thinking assessments in higher education. *Journal of Computing in Higher Education*, *34*(2), 416–461. https://doi.org/10.1007/s12528-021 09305-y

Marques, M., Ochoa, S. F., Bastarrica, M. C., & Gutierrez, F. J. (2018). Enhancing the Student Learning Experience in Software Engineering Project Courses. *IEEE Transactions on Education*, *61*(1), 63–73. https://doi.org/10.1109/TE.2017.2742989

Pei, C. (Yu), Weintrop, D., & Wilensky, U. (2018). Cultivating Computational Thinking Practices and Mathematical Habits of Mind in Lattice Land. *Mathematical Thinking and Learning*, *20*(1), 75–89. https://doi.org/10.1080/10986065.2018.1403543

Rahayu, N. S., Liddini, U. H., & Maarif, S. (2022). Berpikir Kreatif Matematis: Sebuah Pemetaan Literatur dengan

p-ISSN: 2086-4280
e-ISSN: 2527-8827

Analisis Bibliometri Menggunakan Vos Viewer. *Mosharafa: Jurnal Pendidikan Matematika*, *11*(2), 179-190.

Rambally, G. (2017). *Integrating Computational Thinking in Discrete Structures BT - Emerging Research, Practice, and Policy on Computational Thinking* (P. J. Rich & C. B. Hodges (eds.); pp. 99–119). Springer International Publishing. https://doi.org/10.1007/978-3-319-52691-1_7

Rodríguez del Rey, Y. A., Cawanga Cambinda, I. N., Deco, C., Bender, C., Avello-Martínez, R., & Villalba-Condori, K. O. (2021). Developing computational thinking with a module of solved problems. *Computer Applications in Engineering Education*, *29*(3), 506–516. https://doi.org/https://doi.org/10.1002/cae.22214

Rousse, B. S., & Dreyfus, S. E. (2021). Revisiting the Six Stages of Skill Acquisition. *Teaching and Learning for Adult Skill Acquisition: Applying the Dreyfus & Dreyfus Model in Different Fields*, *1980*, 3–30. https://www.infoagepub.com/

Pipitgool, S., Pimdee, P., Tuntiwongwanich, S., & Narabin, A. (2021). Enhancing Student Computational Thinking Skills by use of a Flipped-Classroom Learning Model and Critical Thinking Problem-Solving Activities: A Conceptual Framework. *Turkish Journal of Computer and Mathematics Education*, *12*(14), 1352–1363.

Seehorn, D., Carey, S., Fuschetto, B., Lee, I., Moix, D., Boucher Owens, D., Stephenson, C., & Verno, A. (2011). *CSTA K – 12 Computer Science Standards*.

Selby, C. C. (2015). Relationships: Computational Thinking, Pedagogy of Programming, and Bloom's Taxonomy. *Proceedings of the Workshop in Primary and Secondary Computing Education*, 80–87. https://doi.org/10.1145/2818314.2818315

Selby, C., & Woollard, J. (2013). *Computational thinking: the developing definition*. University of Southampton (E-prints). https://eprints.soton.ac.uk/356481/

Son, Y. S., & Lee, K. J. (2016). Computational thinking teaching model design for activating IT convergence education. *The Journal of the Korea institute of electronic communication sciences*, *11*(5), 511–522. https://doi.org/10.13067/JKIECS.2016.11.5.511

Sung, W., & Black, J. B. (2021). Factors to consider when designing effective learning: Infusing computational thinking in mathematics to support thinking-doing. *Journal of Research on Technology in Education*, *53*(4), 404–426. https://doi.org/10.1080/15391523.2020.1784066

Verenikina, I. (2008). Scaffolding and learning: Its role in nurturing new learners. *Https://Ro.Uow.Edu.Au/Edupapers/43*, *May*.

Wing, J. (2011). *Research notebook: Computational thinking—What and why?* The Link Magazine, June 23, 2015.

Wing, J. M. (2006). Computational Thinking. *Commun. ACM*, *49*(3), 33–35. https://doi.org/10.1145/1118178.1118215

Author's Bibliography

Enjun Junaeti, M.Si.

Born in Majalengka, December 20, 1985. Teaching staff at the Computer Science Education Study Program, FPMIPA UPI. Study S1 Mathematics UPI, Bandung, graduated in 2008; S2 Mathematics ITB, Bandung, graduated in 2011; and is currently pursuing a Doctoral Degree in Mathematics Education at the Indonesian University of Education.

Prof Dr. H. Tatang Herman, M.Ed.

Born in Garut, October 1, 1962. Teaching staff at the Mathematics Education Study Program, FPMIPA UPI. Study S1 Mathematics Education IKIP Bandung, Bandung, graduated in 1989; Master of Mathematics Education at Deakin University, Melbourne, graduated in 1996; and UPI Mathematics Education Doctoral Program, Bandung, graduated in 2006.

Prof. Dr. H. Nanang Priatna, M.Si.

Born in Kuningan, March 31, 1963. Teaching staff at the Mathematics Education Study Program, FPMIPA UPI. Study S1 Mathematics Education IKIP Bandung, Bandung, graduated in 1987; Master of Mathematics Education from IKIP Malang, Malang, graduated in 1994; and Doctor of Mathematics Education UPI, Bandung, graduated in 2003.

Dr. Dadan Dasari, M.Si.

Born in Kuningan, July 17, 1964. Teaching staff at the Mathematics Education Study Program, FPMIPA UPI. Study S1 Mathematics Education IKIP Bandung, Bandung, graduated in 1989; Masters in Mathematics UGM, Yogyakarta, graduated in 1995; and S3 Mathematics Education UPI, Bandung, graduated in 2009.

Dr. Dadang Juandi, M.Si.

Born in Tasikmalaya, January 17, 1964. Teaching staff at the Mathematics Education Study Program, FPMIPA UPI. Study S1 Mathematics Education IKIP Bandung, Bandung, graduated in 1989; Masters in Mathematics UGM, Yogyakarta, graduated in 1997; and S3 Mathematics Education UPI, Bandung, graduated in 2000.