Research Article https://doi.org/10.12973/eu-jer.11.3.1495

European Journal of Educational Research

Volume 11, Issue 3, 1495 - 1507.

ISSN: 2165-8714

http://www.eu-jer.com/

Computational Thinking Process of Prospective Mathematics Teacher in Solving Diophantine Linear Equation Problems

Neneng Aminah*
Semarang State University, INDONESIA

Yohanes Leonardus Sukestiyarno
Semarang State University, INDONESIA

Wardono Wardono
Semarang State University, INDONESIA

Adi Nur Cahyono
Semarang State University, INDONESIA

Received: December 20, 2021 ▪ Revised: March 2, 2022 ▪ Accepted: May 18, 2022

Abstract: Prospective teachers facing the 21st century are expected to have the ability to solve problems with a computer mindset. Problems in learning mathematics also require the concept of computational thinking (CT). However, many still find it challenging to solve this problem. The subjects in this study were twenty-one prospective mathematics teachers who took number theory courses, and then two research samples were selected using the purposive sampling technique. This study uses a qualitative descriptive method to describe the thinking process of prospective teachers in solving Diophantine linear equation problems. The results showed that the first subject's thought process was started by turning the problem into a mathematical symbol, looking for the Largest Common Factor (LCF) with the Euclidean algorithm, decomposition process, and evaluation. The second subject does not turn the problem into symbols and does not step back in the algorithm. The researcher found that teacher candidates who found solutions correctly in their thinking process solved mathematical problem used CT components, including reflective abstraction thinking, algorithmic thinking, decomposition, and evaluation. Further research is needed to develop the CT components from the findings of this study on other materials through learning with a CT approach.

Keywords: *APOS, computational thinking, mathematical problem.*

To cite this article: Aminah, N., Sukestiyarno, Y. L., Wardono, W., & Cahyono, A. N. (2022). Computational thinking process of prospective mathematics teacher in solving Diophantine linear equation problems. *European Journal of Educational Research*, *11*(3), 1495-1507. https://doi.org/10.12973/eu-jer.11.3.1495

Introduction

Currently, technology is growing and impacts all fields, one of which is education. Therefore, activities in learning are always referred to as technology, which impacts the progress of students' mindsets, which Wing (2010) calls computational thinking (CT). Teachers and prospective teachers must be able to integrate CT in classroom learning and the development of CT activities through its components.

Abstraction thinking is a component of CT proposed by many researchers (Barr & Stephenson, 2011; Henderson et al., 2007; Wing, 2010, 2017). Abstraction is an activity that allows us to realize the similarities between experiences that have happened to us. Furthermore, Herskowitz et al. (Skemp 1976, as cited in Mitchelmore & White, 2000) define abstraction as a vertical thinking activity from mathematical concepts previously built by the mind into a new mathematical structure. A series of abstract concept formation activities is an abstraction process, while the main abstraction is the search for the same or general properties of real examples (Hershkovitz et al., 2019). Abstraction has many meanings; extraction, decontextualization, and essence are the most common meanings (Cetin & Dubinsky, 2017). Abstraction thinking is important for students, especially when teaching mathematical concepts in real-world contexts. Abstraction thinking is supported students to create concrete models from abstracts (Gravemeijer, 2011).

In addition, CT skills are also considered as algorithmic thinking that uses principles from computer science as a structural and metaphorical framework (Pala & Mıhçı Türker, 2019). The term is defined as a step-by-step instruction procedure in solving problems, not only in computer science but can be used in other sciences (Selby & Woollard, 2013). The same theme is continued by Sussman (Selby, 2015) who defines CT as a way of explicit instruction to complete a task.

* **Corresponding author:**
Neneng Aminah, Semarang State University, Department of Mathematics, Faculty of Mathematics and Natural Science, Indonesia.
✉ nenengaminah255@students.unnes.ac.id

CT also requires breaking down complex tasks (problems) into smaller, more detailed tasks called decomposition (Lavigne et al., 2020; Selby & Woollard, 2013) *Decomposition* is defined as breaking down into smaller, easier to solve parts (Selby, 2015). Decomposition is a basic skill in problem-solving (K. M. Rich et al., 2018). Selby found that decomposition is a difficult CT skill to master because sometimes the problem to be solved is not fully understood (Selby, 2015). In addition, students seem to understand the concept of decomposition but struggle to apply the process in new situations. The Royal Society (as cited in (P. Rich et al., 2019) states that understanding decomposition is necessary before paying attention to other foundations.

Next is the evaluation process, which generates information from previous experiences (Worthen & Luiselli, 2017; Worthen et al., 2019). The evaluation process is not just measuring the extent to which goals are achieved but is used to make decisions (Cronbach & Stufflebeam, as cited in Arikunto, 2016).

Previous research has carried out various approaches in teaching to solve problems. Researchers in computer programming have proposed CT as an approach to solving problems, but Wing (2010) proposed that CT is not a skill that is only useful for computer scientists but a basic skill that everyone should learn. Increasing technological advances demand CT in every scientific field (Pei et al., 2018; Weintrop et al., 2016). Although there is a clear relevance between CT and computer science, researchers argue that CT should be taught in disciplines outside of computer science starting at an early age (Barr & Stephenson, 2011; Yadav et al., 2016).

Reviewing the investigative studies that have been carried out (Borkulo et al., 2021), the use of conditional questions and GeoGebra software is one way to find CT in solving problems in mathematics learning. Another study conducted (Lee et al., 2011), a research study conducted on high school students, CT, described the use of abstraction, automation, and analysis in problem-solving. The study results present a "Use-modify-create" framework, which represents the three stages of students' cognitive and practical activity in computational thinking. Likewise, research carried out by (Liu & Wang, 2010) with student research subjects in discrete courses, the results of research on computational thinking exercises in solving discrete problems in mathematics t eaching. This research resulted in four main components of computational thinking: abstract thinking, logical thinking, patterned thinking, and constructive thinking. Some of the research that has been disclosed has the advantage that computational thinking can be used in solving problems in learning mathematics and can construct cognitive and practical activities for students. However, it has a weakness that no one has reported the research results on prospective teachers, even though prospective teachers are the most important asset to providing teaching to students.

Our initial observations found that prospective mathematics teachers still had difficulties in solving Diophantine linear equations. They are still looking at existing procedural examples. Furthermore, when given non-routine practical questions, they still have difficulty connecting calculations with theorems. As with previous studies on approaches in teaching to solve problems, researchers reveal that CT as an approach to solving problems in learning (Czerkawski & Lyman, 2015; Liu & Wang, 2010; Rodríguez-Martínez et al., 2019), and Wing (2010) assert that CT is not a skill that is only useful for computer scientists, but a basic skill that everyone should learn.

Therefore, computational thinking skills are needed to solve problems based on theoretical studies and previous research. So, this study reveals the CT components that arise from the analysis of the thinking process of prospective teachers in solving math problems. This research is expected to produce new findings that can be used in further research and help solve problems in learning mathematics

Methodology

Research Design

This study qualitatively describes the thinking process of prospective teachers in solving linear Diophantine equation problems. The characteristics of qualitative research are natural settings, researchers as key instruments, multiple data sources, inductive data analysis, emergent designs, and holistic accounts (Creswell & Plano Clark, 2018).

Sample and Data Collection

The data of this research were taken from the third-semester mathematics teacher candidate. This study involved twenty-one participants as research subjects, consisting of three men and eighteen women aged between 19-21 years. The research subjects were then selected two research samples using purposive sampling technique. Purposive sampling is a technique for sampling data sources with certain considerations (Sukestiyarno, 2020). The concern in this study is to select a sample with the final result criteria by answering true and false. Participants were taken from prospective teachers who have active communication, with the reason to make it easier for researchers to explore information. Taking participants with female gender and 21 years old, because cognitively individuals begin to think interpretively (Santrock, 2011). Furthermore, the subject who answered correctly was named S1, and the subject who answered incorrectly was named S2. This study applied interviews to collect data and activity records using the linear Diophantine equation problem test. Qualitative analysis is carried out by data reduction, data presentation, data interpretation, and concluding.

Expert judgment validated the linear Diophantine equation test instrument. The research instrument was designed according to the purpose. Then, researchers conducted a limited trial of the research instrument. The results of the item validity test show that the five questions are included in the valid and reliable categories. Meanwhile, the experts validated the interview guide to dig up more in-depth information.

Data Collection

Documentation data is in the form of written test results and videos when working on test questions whose implementation is carried out through a zoom meeting with three supervisors. Interview questions were used to clarify the collected test answer sheets (Creswell & Plano Clark, 2018) and strengthen the coding process. This data was selected via cellphone audio, while field notes were research notes during the research process.

The research instrument is in the form of test questions to collect data about the thinking process in solving problems. The instrument was used to collect test data. The instrument was used to collect test data by being given non-routine questions with the following Diopathine linear equation material. *"A factory produces 2400 units every day. The food will be packaged in two types of packaging, each containing five units and 12 units. How many ways are there for possible packing pairs?".*

Data Analysis

To be valid and reliable, the triangulation of data sources used in this study compared the results of the answers to the Diopathine linear equation data and in-depth interviews to find out the truth. This thought process was analyzed using the theoretical analysis tool APOS (action-process-object-schema) to explore the thought process (Arnon et al., 2014; Bintoro et al., 2021). APOS theory explains the development of individual schemas on certain mathematical topics that constitute that knowledge's totality (Dubinsky & Mcdonald, 2001).

The stages of data analysis in this study began with data transfer in solving linear diophantine problems, reducing data, making abstractions, compiling each part of the data, grouping the component data of computational thinking based on coding and ending with concluding. The coding carried out by the researcher is divided into three columns, column one contains the raw data, column two contains the codes per sentence in the raw data, and column three contains the codes of the overall data segment. For this reason, in this study, we provide an explanation using the final code, as shown in table 1.

Table 1. APOS Stage Coding in Process Thinking

Code	Term	Code	Term
	Equation Diophantine		Processes
	Mathematics Models		Object
	LCF		Schema
	Euclidean Algorithm	*Inter-Act*	Interiorization Action
	General Solution	*Matematisation*	Matematization
	Number of Possible Method		Equation Linear Diophantine
	Substitution	*Enca-Pro*	Encapsulation Processes
	Action	*De-Enc*	De-Encapsulation

Findings/Results

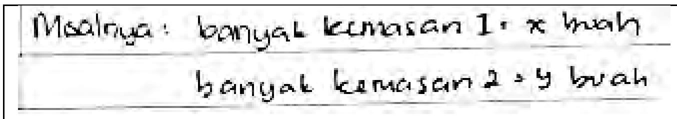
The findings in the first study are that S1 performs a computational thinking process with the concepts of reflective abstraction thinking, algorithmic thinking, decomposition, and evaluation. While the second finding, S2, performs the computational thinking process, when the algorithm's thinking process is disturbed, it does not perform the Euclidean algorithm backwards.

The previous subject had to convert the problem into mathematical symbols, solving the first condition by finding the LCF value of the coefficient on the problem using the steps of the Euclidean algorithm. The subject looks for a specific solution, looks for a general solution, and ends with the process of determining the value to reach a decision, resulting in a solution in the form of integers.

The participants of this study were second-level students (totally 21 people = three males and 19 females) who were taking a number theory course. They were given a written CT test regarding problem-solving in everyday life. The test results were five people (high score = all women) and 16 people (medium score = 3 men and 13 women). To facilitate the implementation of the research, the researcher took two samples, namely one person from each group with high scores (first sample = S1) and moderate (second sample = S2). The categories in this election are female (because there are no men in high scores), can communicate actively, and are 21 years old. The following will explain the thinking process for solving mathematical problems of the selected participants.

Data Analysis of the High Score Group (S1)

Researchers analyzed the S1 answer sheet with support from the results of field notes and interviews. S1 as a whole has the same score as the other subjects. First, we analyzed the action stage in APOS theory. At this stage, S1 understood the problem by describing what is happening in mathematical symbols. S1 represented mathematical ideas in mathematical language and symbols, assuming the number of packages 1 and 2 with symbols x and y. This activity reflects abstract thinking from unstructured abstracts to developed abstracts (Gray & Tall, 2007; Simon, 2020; Toscano, 2008).



Translation:

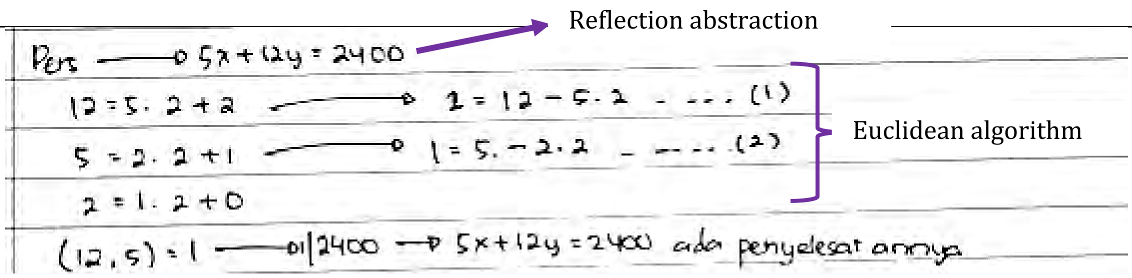
For example:

number of packs 1 = *x* pcs

number of packs 2 = *y* pcs

Figure 1. Abstraction Thinking by S1.

The action stage carried out by S1 is making connections from concepts when encapsulating to form new understandings to find further solutions. Reading, understanding, and conveying it to the brain, and then processing it into developing thoughts are poured into the concept of linear equations. This is seen in the following picture.



Translation:

Equation *5x+12y=2400*

Finding the FPB of *5* and *12*

Because it has a value of *1*, then the next equation has a solution

Figure 2. Algorithm Thinking by S1.

Subject 1 (S1) in the action stage made effort to solve the problem by making linear equations, and interiorization occurred from an action in the form of a change in procedural activity to mental construction in a relative internal process by doing from what is in mind without doing all stages. Since researchers wanted to get more in-depth information, the following interview was conducted.

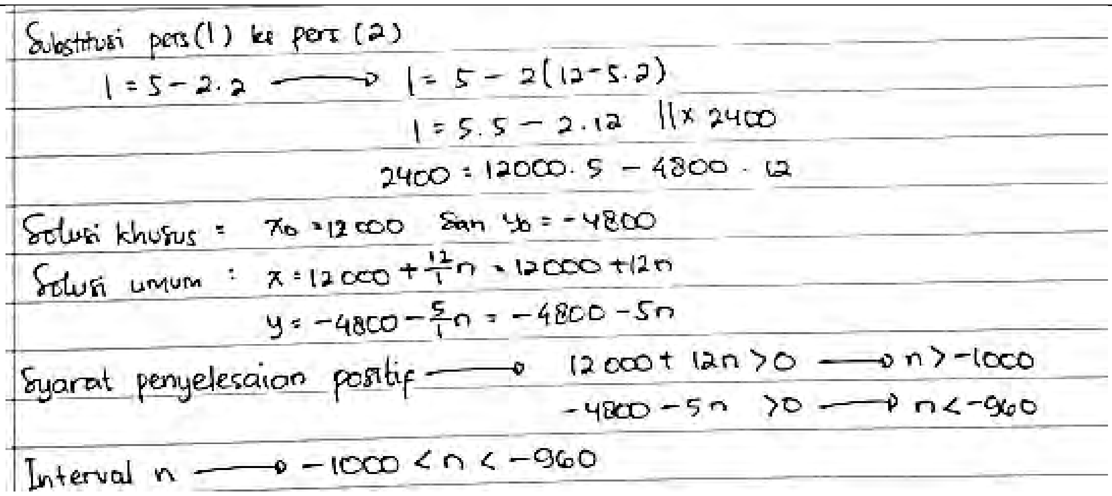
Q: "What do you do after converting something into a mathematical symbol?"

S1: "From making the mathematical symbols x and y, I create a linear equation 5x+12y=2400."

There is a thought process from previous experiences into new experiences by making linear equations from what S1 has done. The thinking process carried out by S1 involves the construction of new knowledge at a higher level. This thinking process is called reflective abstraction thinking (Ferrari, 2003; Simon, 2020).

The process stage is the second stage of APOS theory. S1 prepared a problem-solving plan with regular steps, made a flow to take the next step to find the LCF value from coefficients *12* and *5* using the Euclidean algorithm to produce a value of *1*. If the value obtained can divide the coefficient *c*, the equation has a solution (Figure 2). This kind of thinking process is called algorithmic thinking. The thinking process takes place using the Euclidean algorithm to produce a special solution by performing the backward calculation process on the Euclidean algorithm (Figure 2).

The object stage them takes place. At this stage, the encapsulation process occurs and creates a mental transformation of a process on a cognitive object. This is seen from the indication that S1 reflected on the application of the Euclidean algorithm for the process of looking for linear combinations, and carried out the thought process of genetic decomposition by way of substitution from equation *1* to equation *2* to produce a simpler problem, as shown in Figure 3.



Translation:

Substitute equation (1) into equation (2)

Particular solution: $x_0 = 1200\ and\ y_0 = -4800$

General solution: $x = 1200 + \frac{12}{1}n = 1200 + 12n$; $y = -4800 - \frac{5}{1}n = -4800 - 5n$

Positive settlement conditions $1200 + 12n > 0 \rightarrow n > -1000$;

$$-4800 - 5n > 0 \rightarrow n < -960$$

Interval $n \rightarrow -1000 < n < -960$

Figure 3. Decomposition Thinking Process by S1

Researchers needed truth data, and proceeded with conducting the following interview.

Q: "Where did you get the custom solution from?"

S1: "Before I substituted equations 1 and 2 first, then I had to multiply both sides by a number to produce an equation similar to the original equation."

For the explanation of the interview results, S1 performed a decomposition process which can be seen from the process of substituting equation 1 into equation 2 to produce simpler changes to solve the problem. Figure 3 also explains the third stage of APOS theory, i.e., Object, the cognitive structure stage where students are aware of the transformation processes as a single unit and that transformations can be carried out in a single unit. The object stage of the transformation can be seen from the results of interviews that provide the steps taken to solve the problem. Decomposition occurs at this stage, and it can be seen from the thought process that mental construction occurs by constructing x and y variable objects as actions, which includes doing iterations by substituting special solutions on general solutions. This action was repeated so that the de-encapsulation process occurred. S1 added one backward step process in the Euclidean algorithm to get the hint for the interval value of n.

The schema stage is the last in APOS theory. This is the construction that relates separate actions, processes, objects to certain objects to produce a certain schema. This is seen from the picture below.

Banyak pasangan penyelesaian x & y → 39 buah pasangan x & y.

Jadi, ada 39 cara pasangan kemasan yg mungkin.

Translation:

The number of solutions for x and y is 39 pairs, so there are 39 possible packing pairs

Figure 4. Evaluation Process by S1

The thinking process that occurs at the schema stage (Figure 4) relates to the stages carried out to solve the problem, as evidenced by finding 39 possible packaging pairs. Schemata formed in a person's memory will show knowledge that has been arranged in an interrelated pattern, is built from all previous experiences, and allows to predict the next new knowledge (Do et al., 2003; Tse, 2011; Tse et al., 2007; Van Kesteren et al., 2012). Overall, S1 carries out a thinking process with the concept of computational thinking, which can be seen from the way of thinking, changing the situation into mathematical symbols (abstraction), looking for LCF values and ensuring linear combinations using the Euclidean algorithm (algorithm). Next, S1 raises its genetic decomposition by looking for simpler changes to make it easier to solve the problem. The subject performs the last stage of collecting all the processes they experienced into a meaningful new thing by determining the value for a thing or object based on certain references (evaluation). The stages of action, process, and object to produce a perfect schema can be seen from figure 5 below.

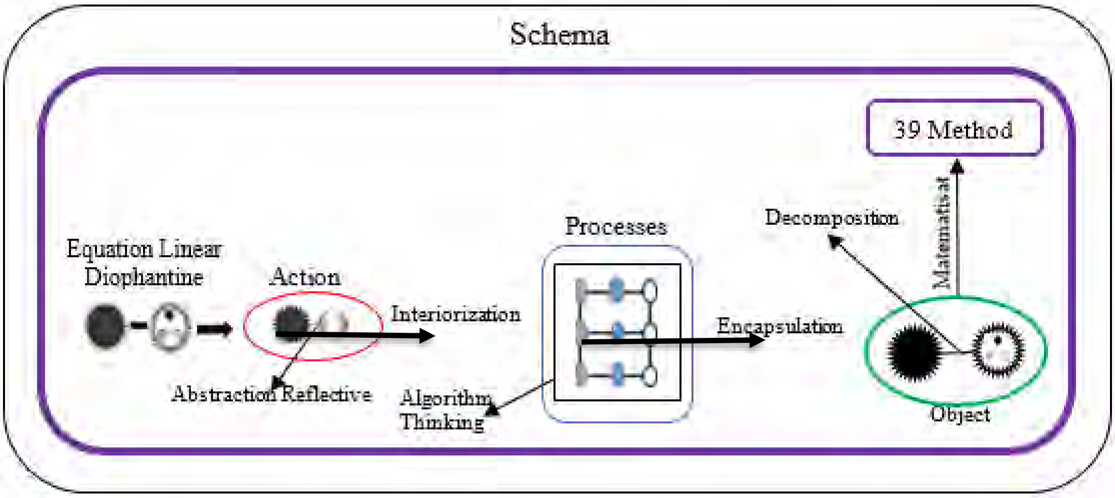


Figure 5. *The Thinking Process Carried Out by S1.*

Figure 5 describes the computational thinking process to produce the desired solution, starting with the cognitive schema abstraction thinking process producing mathematical symbols, and then an algorithmic thinking process occurs, where there is a repeated iteration of thinking using the Euclidean algorithm, the process goes well followed by a countdown on Euclidean algorithm. The object stage occurs by constructing variables x and y to get instructions for generating intervals of n from the previously processed computational thinking process. This is a reference to produce a process in determining meaningful values for decision making.

Analysis of Data from the Moderate Score Group (S2)

S2 describes the thinking process with computational thinking in solving linear Diophantine equation problems through answer sheets. The first stage is the action of S2 carrying out the process of completing the main requirement, i.e., finding the LCF from coefficients 5 and 12 to find out whether to divide the coefficient *c*. This is seen in the following picture.

1). mencari FPB dari 5 dan 12.

12 = 5.2 + 2 → (12,5) = (5,2)

5 = 2.2 + 1 → (5,2) = (2,1)

2 = 1.2 + 0 → (2,1) = (1,0).

↓

FPB.

Translation: Find LCF 5 and 12

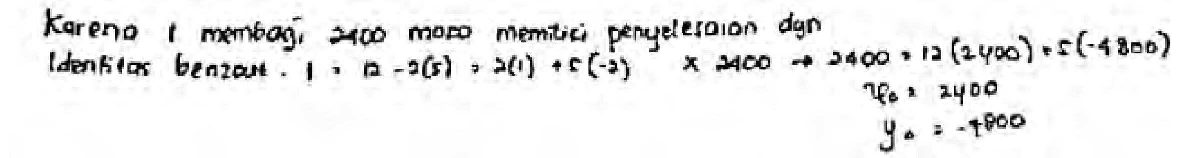
Figure 6. Algorithm Thinking Process by S2

Researchers wanted to get more in-depth information, so the following interview was conducted.

Q: "What are you doing this process for?"

S2: "Before finishing the next step, I have to find the LCF of 5 and 12 using Euclidean algorithm. If it produces a value that is divisible by, then it can solve to the next step."

Subject 2 (S2) at the action stage attempted to solve problems by making connections between processes or concepts to form a new understanding. S2 made the concept relationship a thought process to find LCF using the Euclidean algorithm to get a new understanding to carry out the next process. The process stage is the second stage of APOS theory. S2 performed it with regular steps using Euclidean algorithm. Algorithm performance in life is valuable because it involves many vital activities following simple and discrete steps (Looi et al., 2018; Sys, 2018). However, encapsulation occurs when using Euclidean algorithm, and there is a memory schema error in calling long-term memory. S2 did not perform a step back process on Euclidean algorithm to generate a new understanding (see figure 7).


Karena 1 membagi 2400 maka memiliki penyelesaian dgn
Identitas benzout . $1 = 12 - 2(5) = 2(1) + 5(-2)$ x 2400 → $2400 = 12(2400) + 5(-4800)$
$x_0 = 2400$
$y_0 = -4800$

Translation:

Since *1* divides *2400*, it has a solution

1 = 12 – 2(5) = 2(1) + 5 (-2) u/x = 2400 → *2400 = 12(2400) + 5 (-4800).*

$x_0 = 2400$ and $y_0 = -4800$

Figure 7. De-encapsulation by S2

At this stage, S2 looked for the values of x_0 and y_0 from the identity equation of benzoate, which is a step that must be passed in solving the problem of the Linear Diophantine Equation. It can be seen on the answer sheet that there is a process that should not occur. When writing the benzoate equation, S2 did not do the substitution process from the previous step's results, resulting in the wrong solution. Researchers looked for further data to ensure the flow of the occurred thinking process through an interview with the subject.

Q: "There seems to be an error while making the Benzoate Identity. Can you explain?"

S1: "After I checked, I realized that there was a mistake at this stage. I did not substitute the equation obtained earlier."

From this explanation, researchers found that the thought process was interrupted (Figure 7). There was no good flow of thinking from previous experiences and the loss of long-term memory. It did not produce simple changes to produce a perfect solution (decomposition). The subject should make an equation from the Euclidean algorithm by doing the substitution process from equation 1 to equation 2. The next APOS stage did not go well. This is seen in figure 8 below.

Solusi permasalahan

$x = 2400 + \frac{5}{1}t = 2400 + 5t$

$y = -400 - \frac{12}{1}t = -4800 - 12t$

Maka kemasan yg mungkin dibuat

$\frac{-2400}{5} < t < \frac{-4800}{12}$ atau $-4800 < t < 400$

Jadi nilai $t = -479, -478, -477, \ldots, -400$ ada 1 nilai t

banyaknya kemasan yg dibuat dari 5 unit & 12 unit

adalah 366 u/t $= -479$ maka $x = 5$ dan $y = 474$

u/t $= -400$ maka $x = 400$ dan $y = 0$.

Translation:

Problem solution

$x = 2400 + \frac{5}{1}t = 2400 + 5t$; $y = -400 - \frac{12}{1}t = -400 - 12t$

Then the possible packaging is made $-\frac{2400}{5} < t < -\frac{4800}{12}$ $or - 4800 < t < 400$

So, Score $t = -479, -478, -477, \ldots, -400$ there is 1 score t

The number of packs made of 5 units and 12 units is 366

For *t = -479 so x=5 and y=474*

For *t=-400 so x=400 and y=0*

Figure 8. Evaluations Carried Out by S2.

Figure 8 explains that there was an error in the previous stage so that the scheme was not perfect, so it was wrong at the final stage, the experience from the previous stage was cut off. The flow of S2 thinking is reflected in the following picture.

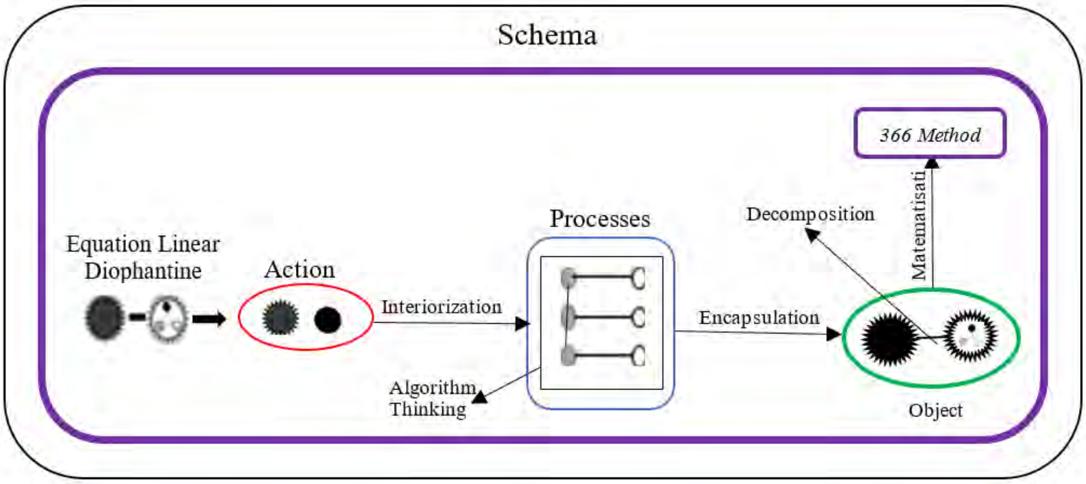


Figure 9 The Thinking Process Carried Out by S2.

Figure 9 describes the flow of thinking using the S2 computational concept. The findings obtained are only the algorithmic thinking process and decomposition. A thought process was interrupted when performing the algorithm, errors at the de-encapsulation stage occurred, and there was no new knowledge to solve the problems S2 faced. S2 did not step back when calculating Euclidean algorithm. S2 lost their old experience, and they should have done the substitution first, so they got an imperfect final score.

Discussion

From the study results, it is known that the thinking process of prospective teachers in solving their problems uses a component of computational thinking. This component was found after going through interviews and written assignment documents that they did. They, when solving problems, carry out a reflective abstraction thought process. As revealed (Cetin & Dubinsky, 2017), reflective abstraction can be an important component in computational thinking. The analysis results reveal that the abstraction focuses on what details we can neglect to think about CT and make the components of CT the most important. Abstract thinking, expressed by (Liu & Wang, 2010), is a method of thinking that solves new problems with abstractions, always translates the source problem into the target problem, and then uses several axioms to solve the problem.

Furthermore, in this study, it is stated that the subject uses a thought process from concrete to abstract, which is called reflective abstraction, because participants are more specific in adulthood where the abstraction formed is reflective. Piaget divided abstraction into empirical and reflective abstraction (Piaget, 1950, as cited in Gray & Tall, 2007). Empirical abstraction creates categories by deriving general characteristics of object classes through observation. He emphasized that in empirical abstraction, the quality of the object being abstracted is already recognized in the same form. At the same time, reflective abstraction is a student activity and explains the construction of new knowledge at a higher level. Knowledge constructed through reflective abstraction is qualitatively different from the knowledge.

S1 shows the ability to anticipate the results of problem-solving and provide arguments for the decisions they make, as the results of previous research, namely that students' reflective abstraction skills are very important in solving problems (Cetin & Dubinsky, 2017; Fuady et al., 2019; Osborne, 2004; Simon, 2020; Toscano, 2008). Meanwhile, in the second participant (S2), the action stage did not provide mathematical symbols and did not perform the algorithm process perfectly. The findings of reflective abstraction thinking are slightly different from the results of researchers (Ioannou & Angeli, 2016; Selby & Woollard, 2014; Wing, 2015), revealing that abstraction thinking is one component of CT, but its type is not specifically disclosed, because the subjects in the study were children.

Genetic decomposition occurs at the process and object stages, a structured collection of mental activities that build blocks (categories) to describe how concepts/principles can be developed in an individual's mind when solving a problem. Subjects do the work through the process of finding the greatest common factor (LCF). S1 runs perfectly and can call the long-term scheme by substituting the first equation into the second equation. S1 looks for conditions that need to complete the steps to solve the problem by proving that LCF is 1. He does the reverse algorithm solution. Meanwhile, S2 cannot remember long-term schemes from previous experiences; S2 does not continue the stage of substituting the first equation into the second equation.

This study looks like thinking algorithms from work documents answer the problem; all subjects perform stages of thinking when solving problems. S1 carries out a perfect process, starting with giving symbols after reading the information, determining the necessary conditions by looking for LCF, the substitution process, and concluding. S2 failed when concluding because there was a process that was interrupted without going through the substitution process.

Both subjects underwent an evaluation process, namely, carrying out a process to generate information from previous experiences (Worthen & Luiselli, 2017; Worthen et al., 2019). They conclude the information obtained previously, but here it is seen that there is a significant difference that S1 gets a perfect final result, while S2 gets an imperfect answer due to the long-term scheme that was interrupted so that S2 did not go through the decomposition process properly.

The findings of this study provide information in solving the problem of Diophantine Linear equations that can be solved perfectly after going through the thinking process of reflective abstraction, decomposition, algorithms, and evaluation. The cause of the error is clear when the decomposition process is interrupted, indicated by not completing the substitution. In contrast to the research results (Susanti & Taufik, 2021), research subjects were government science students who took social statistics courses. Previous researchers focused on analyzing student work to solve statistical problems with computational thinking components, including decomposition, pattern recognition, abstraction, and design algorithms. Previous research provided information on the components of the algorithm design that were carried out by students well so that it was the highest process carried out by students in solving problems, while the decomposition component was the lowest process carried out by students. The cause of errors, in general, is because students are not used to solving in a structured way. Students are accustomed to solving problems by directly substituting values into formulas without first writing down what is known and looking for what is needed in the question first.

Computational thinking processes intersect with mathematical problem-solving abilities, which have the same significant steps as CT proposed by previous researchers, although the concepts are different but have broad similarities. The main CT models define problems, solve problems, and analyze solutions (Litts et al., 2020; Rodríguez-Martínez et al., 2019; Wing, 2010). The stages of problem-solving in mathematics learning are planning, compiling, completing, and re-examining the problem (Polya, 1973, as cited in Sak, 2011). However, CT has a different concept in its activities but has the same three designs.

The study results contribute to further research in the form of proposed computational thinking components, namely reflective abstraction thinking, decomposition, algorithms and evaluation. Slightly different from the research results conducted by (Bocconi et al., 2016), which proposes abstraction, algorithm, automation, decomposition and generalization as components of CT. The CT component resulting from this research can be used to solve problems and become the basis for developing learning in mathematics classes, building CT for prospective teachers to prepare themselves to explore student CT in the schools where they practice so that this research is very well done to support the activities of prospective teachers and is useful for further research. Such research is supported by previous research conducted by Lee et al. (2020) describes the design and implementation of integrated STEM + CT to integrate CT in school programs. The results showed that most students and teachers reacted positively to the curriculum. This study contributed to the integration of CT and the development of CT in students. It also contributes to teachers' professional development regarding CT integration.

Recommendations

Based on the explanation, data analysis, and discussion, this study provides recommendations in solving mathematical problems, can use the components of reflective abstraction thinking, decomposition, algorithmic thinking, and evaluation. For further research, it is necessary to conduct research on the thinking process in other mathematical problems, such as geometry. It is also necessary to create a learning framework by integrating CT-based technology to develop the CT concept from the findings of this study.

Limitations

The problems discussed are limited to describing the thinking process, which is analyzed from the answers of prospective teachers in solving non-routine problems in the Diophantine Linear Equation material using the CT concept.

Acknowledgements

The author wishes to thank the Directorate General of Higher Education, Ministry of Education and Culture, Indonesia. Universitas Negeri Semarang, and Universitas Swadaya Gunung Jati, Cirebon, Indonesia.

Authorship Contribution Statement

Aminah: Conceptualization, design, analysis/ interpretation, writing. Sukestiyarno: Editing/reviewing, critical revision of manuscript, supervision, final approval. Wardono: Editing/reviewing, supervision, final approval. Cahyono: Editing/reviewing, supervision, final approval.

References

Arikunto, S. (2016). *Evaluation basics*. Bumi Aksara Press.

Arnon, I., Cottrill, J., Dubinsky, E., Oktaç, A., Fuentes, S. R., Trigueros, M., & Weller, K. (2014). *APOS Theory*. Springer. https://doi.org/10.1007/978-1-4614-7966-6

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

Bintoro, H. S., Mulyono, Sukestiyarno, Y. L., & Walid. (2021). The spatial thinking process of the field independent students based on action-process-object-schema theory. *European Journal of Educational Research*, *10*(4), 1807-1823. https://doi.org/10.12973/eu-jer.10.4.1807

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing computational thinking in compulsory education-implications for policy and practice.* Publication Office of the European Union. https://doi.org/10.2791/792158

Borkulo, S. V., Chytas, C., Drijvers, P., Barendsen, E., & Tolboom, J. (2021). Computational thinking in the mathematics classroom: fostering algorithmic thinking and generalization skills using dynamic mathematics software. In A. Muhling, M. Armoni, & M. Berges (Ed.), *The 16th Workshop in Primary and Secondary Computing Education (WiPSCE '21)* (pp. 1-9). ACM Digital Library. https://doi.org/10.1145/3481312.3481319

Cetin, I., & Dubinsky, E. (2017). Reflective abstraction in computational thinking. *Journal of Mathematical Behavior*, *47*(3), 70–80. https://doi.org/10.1016/j.jmathb.2017.06.004

Creswell, J. W., & Plano Clark, V. L. (2018). *Core mixed methods design: Designing and conducting mixed methods research approach.* SAGE Publication.

Czerkawski, B. C., & Lyman, E. W. (2015). Exploring issues about computational thinking in higher education. *TechTrends*, *59*(2), 57–65. https://doi.org/10.1007/s11528-015-0840-3

Do, H., Melnik, S., & Rahm, E. (2003). Comparison of schema matching evaluations. In: A.B. Chaudhri, E. Rahm, M. Jeckle, & R. Unland (Ed.), *Web, Web-Services, and Database Systems. NODe 2002. Lecture Notes in Computer Science* (Vol. 2593, p. 1-7). Springer. https://doi.org/10.1007/3-540-36560-5_17

Dubinsky, E. D., & Mcdonald, M. A. (2001). Apos : A constructivist theory of learning in undergraduate mathematics education. *The Teaching and Learning of Mathematics at University Level*, *7*, 275–282. https://doi.org/10.1007/0-306-47231-7_25

Ferrari, P. L. (2003). Abstraction in mathematics. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *358*(1435), 1225–1230. https://doi.org/10.1098/rstb.2003.1316

Fuady, A., Purwanto, Susiswo, & Rahardjo, S. (2019). Abstraction reflective student in problem solving of Mathematics based cognitive style. *International Journal of Humanities and Innovation*, *2*(4), 103–107. https://doi.org/10.33750/ijhi.v2i4.50

Gravemeijer, K. (2011). How concrete is concrete? *Journal on Mathematics Education*, *2*(1), 1–14. https://doi.org/10.22342/jme.2.1.780.1-14

Gray, E. M., & Tall, D. O. (2007). Abstraction as a natural process of mental compression. *Mathematics Education Research Journal*, *19*(2), 23–40. https://doi.org/10.1007/BF03217454

Henderson, P. B., Henderson, P. B., Cortina, T. J., & Wing, J. M. (2007). Computational thinking. *ACM SIGCSE Bulletin, 39*(1), 195-196. https://doi.org/10.1145/1227310.1227378

Hershkovitz, A., Sitman, R., Israel-fishelson, R., Eguíluz, A., Garaizar, P., & Guenaga, M. (2019). Creativity in the acquisition of computational thinking. *Interactive Learning Environments*, *27*(5-6), 628–644. https://doi.org/10.1080/10494820.2019.1610451

Ioannou, I., & Angeli, C. (2016). A framework and an instructional design model for the development of students' computational and algorithmic thinking. In A. Lebrocq, & M. Birt (Ed.), *Mediterranean Conference on Information Systems (MCIS)* (pp. 1–8). AIS Electronic Library. http://aisel.aisnet.org/mcis2016/19

Lavigne, H. J., Lewis-Presser, A., & Rosenfeld, D. (2020). An exploratory approach for investigating the integration of computational thinking and mathematics for preschool children. *Journal of Digital Learning in Teacher Education*, *36*(1), 63–77. https://doi.org/10.1080/21532974.2019.1693940

Lee, I., Grover, S., Martin, F., Pillai, S., & Malyn-Smith, J. (2020). Computational thinking from a disciplinary perspective: integrating computational thinking in K-12 science, technology, engineering, and mathematics education. *Journal of Science Education and Technology, 29*(1), 1-8. https://doi.org/10.1007/s10956-019-09803-w

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., & Werner, L. (2011). Computational thinking for youth in practice. *Associate Computing Machinery, 2*(1), 32-37. https://doi.org/10.1145/1929887.1929902

Litts, B. K., Lewis, W. E., & Mortensen, C. K. (2020). Engaging youth in computational thinking practices through designing place-based mobile games about local issues. *Interactive Learning Environments*, *28*(3), 302–315. https://doi.org/10.1080/10494820.2019.1674883

Liu, J., & Wang, L. (2010). Computational thinking in discrete mathematics. *Education Technology and Computer Science Computational*, *42*(3), 413–416. https://doi.org/10.1109/ETCS.2010.200

Looi, C. K., How, M. L., Longkai, W., Seow, P., & Liu, L. (2018). Analysis of linkages between an unplugged activity and the development of computational thinking. *Computer Science Education*, *28*(3), 255–279. https://doi.org/10.1080/08993408.2018.1533297

Mitchelmore, M. C., & White, P. (2000). Development of angle concepts by progressive abstraction and generalisation. *Educational Studies in Mathematics*, *41*(2), 209–238. https://doi.org/10.1023/A:1003927811079

Osborne, P. (2004). The reproach of abstraction. *Radical Philosophy*, *127*(2), 21–28. https://bit.ly/3NnQdji

Pala, F. K., & Mıhçı Türker, P. (2019). The effects of different programming trainings on the computational thinking skills. *Interactive Learning Environments*, *20*(4), 1–11. https://doi.org/10.1080/10494820.2019.1635495

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

Rich, K. M., Binkowski, T. A., Strickland, C., & Franklin, D. (2018). Decomposition: A K-8 computational thinking learning trajectory. In A. Korhonen, L. Malmi, R. McCartney (Ed.), *ICER 2018 - Proceedings of the 2018 ACM Conference on International Computing Education Research* (p.124–132). ACM Digital Library. https://doi.org/10.1145/3230977.3230979

Rich, P., Rich, P. J., Egan, G., & Ellsworth, J. (2019). A framework for decomposition in computational thinking. In B. Scharlau, & R. McDermott (Ed.), *ITiCSE '19: Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, (pp. 416–421). Association for Computing Machinery. https://doi.org/10.1145/3304221.3319793

Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-lópez, M. (2019). Computational thinking and mathematics using Scratch : An experiment with sixth-grade students. *Interactive Learning Environments*, *20*(3), 1–12. https://doi.org/10.1080/10494820.2019.1612448

Sak, U. (2011). Selective problem solving (SPS): A model for teaching creative problem-solving. *Gifted Education International*, *27*(3), 349–357. https://doi.org/10.1177/026142941102700310

Santrock, J. W. (2011). *Life-span development*. Erlangga.

Selby, C. (2015). Relationships : Computational thinking , pedagogy of programming , and Bloom ' s Taxonomy. In J. Ezer, J. Vahrenhol, & S. Sentence (Ed.), *WiPSCE '15: Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 80–87). Association for Computing Machinery (ACM). https://doi.org/10.1145/2818314.2818315

Selby, C., & Woollard, J. (2013). Computational thinking : The developing definition. In A. Clear, B. Manaris, & D. Finkel (Ed.), *ITiCSE '13: Proceedings of the 2013 conference on Innovation & technology in computer science education* (pp. 5–8). Association for Computing Machinery (ACM). https://bit.ly/3Mkhovn

Selby, C., & Woollard, J. (2014, December 10). *Refining an understanding of computational thinking*. University of Southampton Institutional Repository. https://eprints.soton.ac.uk/372410/

Simon, M. A. (2020). Elaborating reflective abstraction for instructional design in mathematics: Postulating a second type of reflective abstraction. *Mathematical Thinking and Learning*, *22*(2), 162–171. https://doi.org/10.1080/10986065.2020.1706217

Sukestiyarno, Y. (2020). *Research educational method* (1st ed.). UNNES Press.

Susanti, R., & Taufik, M. (2021).Analysis of students computational thinking in social statistic problem. *Supremum Journal of Mathematics Education, 5*(1), 22-31. https://doi.org/10.35706/sjme.v5i1.4376

Sys, M. M. (2018). From algorithmic to computational thinking : On the way for computing for all students. In V. Dagiene (Ed.), *ITiCSE '15: Proceedings of the 2015 Conference on Innovation and Technology in Computer Science Education* (p. 1-8). Association for Computing Machinery (ACM). https://doi.org/10.1145/2729094.2742582

Toscano, A. (2008). The culture of abstraction. *Theory, Culture & Society*, *25*(4), 57–75. https://doi.org/10.1177/0263276408091983

Tse, D. (2011). Schema dependent gene activation and memory encoding in neocortex. *Science*, *333*(12), 891–895. https://doi.org/10.1126/science.1205274

Tse, D., Langston, R. F., Kakeyama, M., Bethus, I., Spooner, P. A., Wood, E. R., Witter, M. P., & Morris, R. G. M. (2007). Schemas and memory consolidation. *Science*, *316*(4), 76–82. https://doi.org/10.1126/science.1135935

Van Kesteren, M. T. R., Ruiter, D. J., Fernandez, G., & Henson, R. N. (2012). How schema and novelty augment memory formation. *Trends in Neurosciences*, *35*(4), 211–219. https://doi.org/10.1016/j.tins.2012.02.001

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147. https://doi.org/10.1007/s10956-015-9581-5

Wing, J. M. (2010). Computational thinking. *Communications of the ACM*, *49*(3), 33-35. https://dl.acm.org/doi/10.1145/1118178.1118215

Wing, J. M. (2015). Computational thinking and thinking about computing. *Philosophical Transactions of The Royal Society A Mathematical Physical and Engineering Sciences*, *366*(2), 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Wing, J. M. (2017). Computational thinking ' s influence on research and education for all Influenza del pensiero computazionale nella ricerca e nell ' educazione per tutti. *Italian Journal of Educational Technology*, *25*(2), 7–14. https://doi.org/10.17471/2499-4324/922

Worthen, D., & Luiselli, J. K. (2017). Social validity assessment and intervention evaluation of mindfulness education and practices with high school students. *Mindfulness*, *8*(4), 903–910. https://doi.org/10.1007/s12671-016-0664-z

Worthen, M., Veale, A., McKay, S., & Wessells, M. (2019). The transformative and emancipatory potential of participatory evaluation: Reflections from a participatory action research study with war-affected young mothers. *Oxford Development Studies*, *47*(2), 154–170. https://doi.org/10.1080/13600818.2019.1584282

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all: Pedagogical approaches to embedding 21st century problem solving in K-12 classrooms. *TechTrends*, *42*(2), 565–568. https://doi.org/10.1007/s11528-016-0087-7

Appendix

Point of Interview Question Computational Thinking Processes in Solving Mathematics Problem

No	Concept Computational Thinking	Definition	Question
1	*Abstraction thinking*	• It is a process of making artifacts easier to understand by reducing unnecessary detail and number of variables, and leading to easier solutions (Humpreys, 2015) • Thinking that involves the process of transferring similar problems to problem-solving (Bocconi et al. 2016)	How is the process done at the beginning of reading math problems?
2	*Algoritm thinking*	• step-by-step sequence in a job (Bocconi et al. 2016) • involves using a regular sequence of steps to solve a problem or complete a task (Wing J., 2011) • step-by-step instruction set (Barr & Stephenson, 2011)	What are the steps involved in solving the problem?
3	*Decomposition*	The ability to break down complex tasks (problems) into smaller, more detailed tasks (Bocconi et al. 2016; Wing J., 2008).	How to break complex tasks into light tasks?
4	*Evaluation*	think of ways to predict in order to get a possible outcome as a solution. The process of generating information from previous experience (Worthen & Luiselli, 2017; Worthen et al., 2019).	How to predict possible outcomes as a settlement?