LUMAT
International Journal on Math, Science and Technology Education

Research article

Mathematics teachers' perceptions of computational thinking and its integration into mathematics education

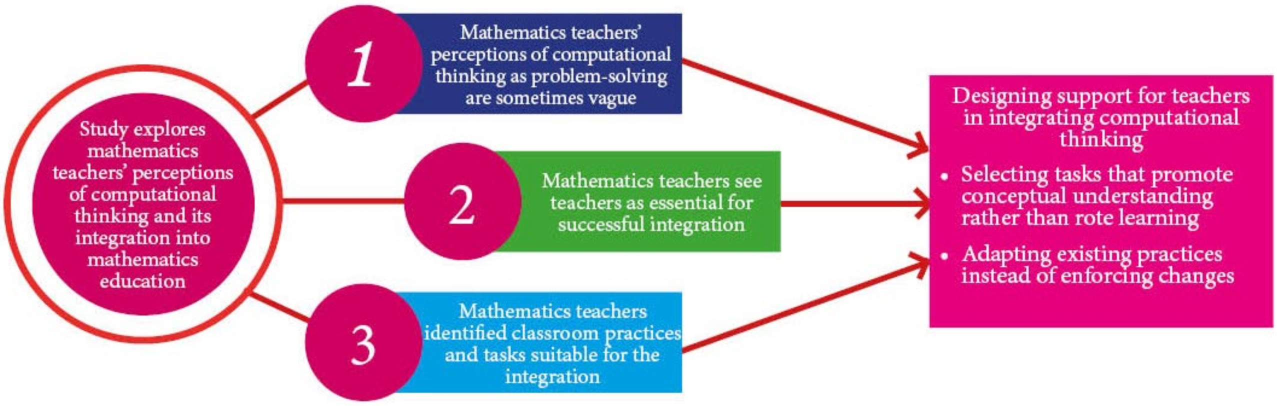
Muhammad Zuhair Zahid

Umeå Mathematics Education Research Centre, Department of Science and Mathematics Education, Umeå University, Sweden

Abstract: Computational thinking is a problem-solving process involving abstraction, algorithmic thinking, automation, debugging, decomposition, and generalisation. It has been increasingly regarded as an essential skill and many countries have attempted to include it in educational systems. In Indonesia, as well as including computational thinking as part of the Informatics subject, the government has encouraged its integration into various subjects, which requires teachers to have a clear and shared understanding of the concept. Thus, this study explores Indonesian mathematics teachers' perceptions of computational thinking and its potential incorporation into teaching and learning mathematics. Semi-structured interviews were used to obtain rich insights. The findings reveal that mathematics teachers have oversimplified perspective on some components such as algorithmic thinking and automation, contributing to their vague perception of computational thinking as problem-solving, which may hinder the original purpose of integrating computational thinking into mathematics education. They also show that the teachers recognise their own importance for successful integration and that existing classroom practices and mathematics tasks can be used for integrating computational thinking into mathematics education. This study contributes to the literature on how teachers conceptualise computational thinking within mathematical domain, situated in the evolving educational context where the integration of computational thinking is still emerging. The study suggests that several factors related to teachers' perceptions of computational thinking should be considered in professional development programmes to support its integration. These include focusing on encouraging teachers to select appropriate mathematics tasks that promote effective computational thinking integration and enhance their current teaching practices with computational thinking.

Keywords: computational thinking, mathematics teachers, teachers' perceptions, integration of computational thinking

Correspondence: zuhair.zahid@umu.se



1 Introduction

The increasing permeation of computing in everyday life has led to growing awareness of the need to foster computational thinking (CT) in schools. CT refers to the conceptual and practical aspects of computer science education (Rich et al., 2019) and now is increasingly regarded as a crucial skill globally. In response, numerous countries, including Indonesia, have initiated efforts to integrate CT into K-12 education (Bocconi et al., 2016; Elicer et al., 2023; Ministry of Education and Culture of Indonesia [MoEC], 2020b). As well as incorporating CT in the formal curriculum through a compulsory subject, Informatics, the Indonesian government has also promoted CT as a relevant skill that can be incorporated into subjects, including mathematics (Dagienė et al., 2022; MoEC, 2019, 2020; SEAMEO Regional Open Learning Center, 2023; Suwaji et al., 2020).

Mathematics is often suggested as a vehicle to introduce students to CT (Weintrop et al., 2016), partly because mathematics and computer science are closely connected: computational methods are increasingly applied in mathematical research (Bailey & Borwein, 2011; Wing, 2008), while mathematics provides theoretical foundations for advances in computer science (Baldwin et al., 2013). From a pedagogical perspective, there is a reciprocal relationship between them; mathematics provides environments and contexts for problems that can be solved by applying CT and involving computational methods in learning mathematics may improve students' understanding of mathematical concepts (Weintrop et al., 2016).

Teachers play an important role when CT is integrated into mathematics as its inclusion depends largely on their perceptions of CT and how they associate the established classroom practices with CT concepts and terms. In addition, it is essential to understand teachers' concerns regarding factors that need to be anticipated for the successful inclusion of CT in mathematics education. Therefore, an essential step in preparing robust support for teachers in bringing CT to mathematics classrooms is to understand their conceptions of CT and their concerns about factors that need to be anticipated when CT becomes part of everyday teaching and learning. Furthermore, most studies on teachers' perceptions of CT and its integration into mathematics have been conducted in Western contexts, with limited research exploring how CT is interpreted in different educational settings. This study addresses this gap by providing insights into how Indonesian mathematics teachers perceive CT and its components, and how they anticipate integrating it into their mathematics instructions.

2 Background

This section starts by reviewing CT and its inclusion in the Indonesian educational system. It then describes recognised components of CT and understandings of these components in mathematics and mathematics education. Finally, it briefly summarises previous studies on teachers' perceptions of CT.

2.1 Computational thinking and attempts to integrate it into Indonesian education

The history of CT can be traced back to the emergence of computer science (CS), when pioneers such as Perlis, Knuth, and Djikstra recognised and discussed the need for a mental framework that could help efforts to solve problems in the CS field (Denning, 2017). Papert (1980) was arguably the first researcher to introduce the term CT but he did not attempt to define it. A major contributor to the growing interest in CT, Wing (2006, 2008, 2011), recognised that computational methodology has already transformed diverse disciplines, including biology, social science, and economics. She introduced the notion of CT as a universally applicable attitude and skill set that could be beneficial for everyone, not just computer scientists, by drawing on fundamental concepts of CS to help efforts to solve problems, design systems, and understand human behaviour (Wing, 2006, 2008).

The idea of utilising core concepts of CS and applying them in other subjects has drawn attention from researchers and policymakers who are interested in incorporating CS into K-12 education. Teaching CS through CT shifts the focus from merely teaching programming, as it involves elements such as: problem formulation; organising, analysing and representing data in new forms; and automating solutions (D. Barr et al., 2011). Furthermore, many aspects of CT such as collaborative and creative problem-solving skills, align with central elements of 21st century skills (Nouri et al., 2020). This has encouraged governments of many countries, including Indonesia, to incorporate CT into education in order to prepare citizens with the digital literacy needed for the future (Bocconi et al., 2016; Hsu et al., 2019; MoEC, 2020).

CT has been formally introduced into Indonesia's educational system mainly as part of the Informatics subject. Informatics was an elective subject in lower and upper secondary school from 2018 (with its implementation depending on schools' conditions and resources) until 2024, when it became a mandatory subject at the secondary level (Regulation of MoEC No. 35 of 2018, 2018; Regulation of MoEC No. 36 of 2018, 2018; Regulation of MoECRT No. 12 of 2024, 2024). The curriculum now treats CT as a foundational element for learning Informatics, and practical aspects of the Informatics subject are supposed to be applied across disciplines (MoEC, 2019, 2020).

As teachers play key roles in successful teaching of CT, trainings are made available for them through various programmes. The government-related initiatives included introducing teachers to CT as one of the aspects evaluated in the Programme for International Student Assessment (PISA) (Suwaji et al., 2020). In addition to these, there were professional development programmes linked to Bebras Indonesia[1] (e.g., Supatmiwati et al., 2024; Wonohadidjojo et al., 2021), along with emerging initiatives led by university educators (e.g., Susilowati et al., 2025; Trisnapradika et al., 2024) and regional education centres (e.g., SEAMEO QITEP in Mathematics, 2021). Yet, published evaluations on their impact, particularly in teaching and learning mathematics at the K-12 level, remain limited. Furthermore, some studies have reported various approaches to CT integration in mathematics, including programming with Scratch (Prahmana et al., 2024), GeoGebra (Yunianto, El-Kasti, et al., 2024), integrated STEAM activities (Yunianto et al., 2025), and spreadsheet (Yunianto, Bautista, et al., 2024). However, these studies tend to be exploratory and lack insights into teachers' conceptualisation of CT. The present study addresses that gap by examining teachers' perception of CT, with particular attention to how they relate its components with mathematics. These CT components are explored in the following section.

2.2 Components of computational thinking

Researchers widely acknowledge CT's importance in educational systems. Definitions and recognised components have varied, but some definitions have overlapping elements. Wing (2006, 2008) initially identified abstraction and automation as key elements and subsequently added problem formulation (Wing, 2011). V. Barr and Stephenson (2011) outlined CT concepts and capabilities including dealing with data, decomposing problems, abstraction, use of algorithms, parallelisation, and simulation. Lee et al. (2011) explored the concept of CT "for youth in practice" and proposed that it should include abstraction, automation, and analysis. In teaching guidance for integrating CT in schools, Csizmadia et al. (2015) held that it encompasses algorithmic thinking, decomposition, pattern identification (generalisation), abstraction, and evaluation. Anderson (2016) suggested that CT consists of decomposition, pattern recognition, abstraction, automation, algorithm design, and evaluation, while Shute et al. (2017) concluded that CT has six main facets: abstraction, decomposition, algorithms, debugging, iteration, and generalisation.

Bocconi et al. (2016) identified constituents of CT by analysing terms that consistently appeared in prominent papers about CT. They described CT as "thought processes entailed in formulating a problem so as to admit a computational solution" and argued that it

[1] Bebras is an international initiative that promotes Informatics and CT for students from grade 3-12. Bebras Indonesia is a National Bebras Organisation (NBO) representing Indonesia in Bebras International community (Dagienė et al., 2022; Natali et al., 2023)

involves abstraction, algorithmic thinking, automation, decomposition, debugging, and generalisation. They based descriptions of abstraction, algorithmic thinking, decomposition, debugging and generalisation on conceptualisations of Csizmadia et al. (2015), and automation on a conceptualisation of Lee et al. (2011). The following text summarises the descriptions by Bocconi et al. (2016), while both the rationale for using them and their application in the study are detailed in the Method section.

2.2.1 Abstraction

Referring to Csizmadia et al. (2015), Bocconi et al. (2016) described abstraction as follows:

> "The process of making an artefact more understandable through reducing the unnecessary detail. The skill in abstraction is in choosing the right detail to hide so that the problem becomes easier, without losing anything that is important. A key part of it is in choosing a good representation of a system. Different representations make different things easy to do" (p.18).

2.2.2 Algorithmic thinking

Algorithmic thinking was described as "a way of getting to a solution through a clear definition of the steps" (Csizmadia et al., 2015, as cited in Bocconi et al., 2016, p. 18).

2.2.3 Automation

Automation, the only component not defined by Csizmadia et al. (2015), was described by Bocconi et al. (2016), following Lee et al. (2011), as "a labour saving process in which a computer is instructed to execute a set of repetitive tasks quickly and efficiently compared to the processing power of a human. In this light, computer programs are "automations of abstractions" (p.18).

2.2.4 Decomposition

Bocconi et al. (2016), again citing Csizmadia et al. (2015), described decomposition as:

> "a way of thinking about artefacts in terms of their component parts. The parts can then be understood, solved, developed and evaluated separately. This makes complex problems easier to solve, novel situations better understood and large systems easier to design" (p.18).

2.2.5 Debugging

Referring to Csizmadia et al. (2015), Bocconi et al. (2016) explained debugging as "the systematic application of analysis and evaluation using skills such as testing, tracing, and logical thinking to predict and verify outcomes" (p.18).

2.2.6 Generalisation

Citing Csizmadia et al. (2015), Bocconi et al. (2016) explained generalisation as:

> “Generalisation is associated with identifying patterns, similarities and connections, and exploiting those features. It is a way of quickly solving new problems based on previous solutions to problems, and building on prior experience. Asking questions such as “Is this similar to a problem I’ve already solved?” and “How is it different?” are important here, as is the process of recognising patterns both in the data being used and the processes/strategies being used. Algorithms that solve some specific problems can be adapted to solve a whole class of similar problems” (p.18).

Some of the CT components mentioned by Bocconi et al. (2016) are already familiar within mathematics and mathematics education. The following section outlines how these CT components are widely recognised in these fields.

2.3 Computational thinking components recognised in mathematics and mathematics education

The CT components described by Bocconi et al. (2016) are not only acknowledged in CS but are also widely applied in other subjects. Some components such as abstraction, algorithmic thinking, decomposition, and generalisation have been widely recognised in mathematics, and each has a special meaning in the context of teaching and learning mathematics.

Abstraction is well known in both CS and mathematics. From a CS perspective, abstraction is regarded as the core of CT (Wing, 2006, 2008). In daily life, ‘abstract’ is commonly understood as unreal, not concrete, or meaningless (Mitchelmore, 2002). This notion is often used in mathematics education to describe the formation of mathematical concepts from sets of contexts to abstract concepts in the form of symbols (Mitchelmore, 2002). In school mathematics, abstraction is associated with solving mathematical word problems, through steps including extraction of numeric information from a narrative and recognising the underlying mathematics that should be addressed (Schley & Fujita, 2014). Rich and Yadav (2020) also argued that teaching abstraction in CS shares similarities with teaching word problems in mathematics, and thus advocated the adoption of good practices from teaching abstraction in CS to enhance the teaching of ways to address word problems in mathematics.

Algorithmic thinking has been linked to mathematical thinking. Knuth (1985) attempted to distinguish algorithmic and mathematical thinking, concluding that both types of thinking share several modes of thought and, therefore, cannot be separated entirely. Mathematicians who participated in a more recent study by Lockwood et al. (2016) highlighted the sequential nature of steps in algorithmic thinking and associated it with procedural knowledge in mathematics.

Moreover, decomposition and generalisation were both mentioned by Polya (2004) as strategies for mathematical problem-solving. According to Polya, decomposition includes breaking a whole into its parts, working on each part, recombining the parts into the whole and carefully considering which parts of the problems are known and which are not. Decomposition is also recognised as a strategy for doing arithmetic operations that is related to *part-part whole* reasoning and useful, *inter alia*, for helping children to solve addition problems at an early age (Cheng, 2012). Similarly, generalisation has been recognised as essential for solving some problems (Polya, 2004), for example by looking for common features of mathematical objects (Dörfler, 1991). Generalisation in mathematics is often closely linked to pattern recognition (Zazkis & Liljedahl, 2002). In school mathematics early algebra is also frequently viewed as the generalisation of familiar arithmetic (Kaput, 2017).

The points discussed above highlight connections between CT components, mathematics, and mathematics education that have also been recognised by educators, who have related CT and classroom mathematics, as discussed in the following subsection.

2.4 Mathematics teachers' perceptions of computational thinking in mathematics

A commonly reported perception among mathematics teachers is that CT in mathematics primarily involves problem-solving skills. Rich et al. (2019) examined American elementary teachers' perceptions of the relation between CT and practices when teaching mathematics and science, through interviews, and found that the teachers recognised CT as a form of problem-solving with strong connections to mathematics. Similarly, Nordby et al. (2022) found, through observations and interviews, that Norwegian primary teachers most closely connected CT with problem-solving and pattern recognition. A Delphi study by Kallia et al. (2021) found that Dutch mathematicians, computer scientists, and secondary teachers characterised CT in mathematics education in terms of problem-solving, cognitive processes, and transposition (i.e., framing a mathematical solution for use by another person or machine). In addition, Huang et al. (2021) investigated Singaporean teachers of grades 7-12 who teach computing and mathematics and found that the teachers perceived CT as a problem-solving approach in both subjects differently. The participants in this study characterised problem-solving in computing as constructing code to demonstrate the solution process, while in mathematics, it was framed as producing a final, computed output supported by reasoning. Taken together, the aforementioned studies suggest that problem-solving is central to how many mathematics teachers understand CT.

Furthermore, some studies have found that although CT is relatively new to teachers they can easily recognise its concepts and tend to link them with practices in mathematics classrooms. Rich et al. (2019) reported that the elementary teachers who participated in their study were familiar with some CT components described by Bocconi et al. (2016), and suggested that this should be taken into account when designing professional development programmes for teachers. Similarly, Humble and Mozelius (2023) found that grade 7-12 mathematics and technology teachers recognised that aspects of their classroom practices aligned with various facets of CT. This reflects one of the views of CT expressed by teachers in the study of Nordby et al. (2022), that CT in mathematics is already covered by aspects currently present in everyday classroom activities. As mentioned in the previous section, CT and mathematics share terms such as abstraction, problem decomposition, and algorithmic thinking, which may help explain why mathematics teachers are generally familiar with CT. However, this overlapping vocabulary may also invite conceptual misunderstandings that hinder their engagement with CT.

The common notions shared by CS and mathematics can create confusion among teachers due to inter-disciplinary and contextual variations in their interpretations. For example, Humble and Mozelius (2023) and Nordby et al. (2022) found that teachers encountered challenges in understanding algorithmic thinking, partly because the term 'algorithmic' often refers to following standard procedures to solve routine problems in mathematics, rather than thinking algorithmically, as in CT. Similarly, in Huang et al.'s (2021) study, teachers' perceptions of algorithmic thinking differ between computing and mathematics: in computing, it was framed as "creating an algorithm", whereas in mathematics, it was interpreted as "following an algorithm". Additionally, several studies have reported that teachers may misunderstand CT as being primarily about using computers. For instance, Marom's (2023) study on Indonesian primary school teachers found that, despite their positive impressions of adopting CT, they perceived it as being highly dependent on the use of computer programs. This finding echoes those of Ling et al. (2017), who reported that when CT was initially introduced, many Malaysian primary teachers regarded it as synonymous with using computers in the classroom. Besides these misconceptions, which stem from shared notions and the association of CT with the use of computers, teachers also acknowledge challenges that may arise when incorporating CT into teaching and learning mathematics.

Teachers express various concerns about integrating CT into their teaching. These include doubts about whether CT is appropriate for elementary students' development level (Rich et al., 2019). They also highlighted their lack of knowledge, including uncertainty about what CT entails (Nordby et al., 2022), how to integrate it into their teaching (Marom, 2023; Stupurienė et al., 2024), and how to assess CT when it becomes part of learning objectives (Ukkonen et al., 2024). Other concerns include inadequate school infrastructure to support the integration, limited time for professional development (Ling et al., 2017; Stupurienė et al., 2024), and the difficulty of convincing colleagues to adopt CT in their teaching practices (Humble & Mozelius, 2023). Taken together, these concerns point to the need for a deeper exploration of how mathematics teachers perceive and navigate CT in mathematics education.

3 Research questions

As mentioned earlier, this study aims to obtain insights into mathematics teachers' perceptions of CT and its integration into mathematics education. The notion of *teachers' perceptions* in this study refers to the ways in which teachers regard, understand, and interpret CT. These perceptions are examined by exploring both their general views of CT and how they associate its individual components (abstraction, algorithmic thinking, automation, debugging, decomposition, and generalisation) with mathematics. Furthermore, as CT is increasingly recognised as relevant across subject areas, this study also explores how teachers perceive the potential integration of CT into mathematics teaching and learning, including their identification of associated opportunities and challenges. The following research questions (RQs) guide this study:

- RQ1: How do mathematics teachers perceive CT, and how do they associate its components with mathematics?
- RQ2: How do they view the potential CT integration in teaching and learning mathematics?

4 Method

Semi-structured interviews were chosen to address the research questions due to their suitability for exploring informants' views in depth and drawing connections between their ideas (Cohen et al., 2018). The participants, interview design, data collection, and analytical processes are described in detail in the following sub-sections.

4.1 Selection of participants

This study follows Cohen et al.'s (2018) suggestion that recruiting knowledgeable participants in a study may help researchers acquire in-depth information about a phenomenon. Thus, in this study participants were recruited through purposeful sampling, targeting mathematics teachers who had participated in one of the government's programmes for CT introduction, as they were likely to have relevant knowledge about CT. Thirteen teachers who had participated in one of the government's introductory programmes were invited to participate in the interviews. Invitations were sent via multiple channels, including email, social media, and personal contacts. Nine teachers (six males and three females) eventually accepted the invitation, with four teaching lower secondary (grades 7—9) and five teaching upper secondary (grades 10—12). The participants' teaching experience varied from 1 to 17 years. All of them had a bachelor's degree in either mathematics or mathematics education. One had a master's degree in mathematics education, one had a master's degree in mathematics, and one had a master's degree in science education. Each teacher received compensation of 100,000 IDR in the form of e-money for participating in the interviews. Table 1 summarises information about the participants.

Table 1. Summary information about participants in the study

Pseudoname	Teaching experience	Educational background	Province where teachers were based when interviewed
Paul	(not known)	Bachelor (Mathematics) and Master (Science Education)	Jambi
Erick	11 years	Bachelor (Mathematics)	East Java
Alex	12 years	Bachelor (Mathematics)	East Java
Andi	1 year	Bachelor (Mathematics Education)	Yogyakarta
Felix	17 years	Bachelor (Mathematics)	East Java
Hanna	(not known)	Bachelor (Mathematics Education) and Master (Mathematics)	South Sulawesi
Robert	6 years	Bachelor (Mathematics Education)	Papua
Heli	9 years	Bachelor (Mathematics Education) and Master (Mathematics Education)	Yogyakarta
Dina (excluded)	(not known)	Bachelor (Mathematics Education)	East Java

4.2 Interview design

The interviews were semi-structured, featuring predetermined questions with flexible wording and sequences tailored to each response. Each interview had three parts: open questions about CT, a discussion of the six CT components, and questions about its potential integration into mathematics instruction.

As mentioned in the background section, CT definitions and components vary, and the participants may have had different perceptions of the components. Therefore, a description of CT and its components is essential to guide the interview process and maximise the consistency of the collected data. Furthermore, as the study involves mathematics teachers at lower and upper secondary levels, selecting a description of CT that might be familiar to the participants is important. The description by Bocconi et al. (2016) was eventually adopted over other frameworks that focus merely on high school settings (e.g., Weintrop et al. 2016) and those situated within the field of CS (e.g., Brennan & Resnick, 2012; Computer Science Teachers Association (CSTA), n.d.; International Society for Technology in Education (ISTE), n.d.). The CT description by Bocconi et al. (2016) aligns with guidelines for teaching CT in schools, and thus was deemed suitable for helping teachers to grasp the intended meanings during the interviews.

Each interview started by asking an open question about the respondents' knowledge of CT. If they touched on some of the six components during responses to the open question, the conversation continued by discussing their knowledge of those components and how they connected it with their teaching. After all CT components mentioned by the participants had been discussed, the interview continued by showing the components mentioned by Bocconi et al. (2016)—abstraction, algorithmic thinking, automation, debugging, decomposition, and generalisation—and discussing the components that had not been discussed during the previous conversation. If the participants did not know about a certain component, they were shown the definition of that component together with some examples, before the discussion continued. When all six components had been addressed, the interview continued to the next session on the potential integration of CT into mathematics lessons. This part included questions about possible benefits of integration, what activities could be designed for that purpose, and what factors could be challenging. Appendix A provides more information about the interview guidelines.

4.3 Data collection

The interviews were conducted online through Zoom meetings from November 2021 to March 2022, following the completion of several CT training programmes in Indonesia. Written consent was obtained from all nine participants. They were also verbally informed before each interview that the Zoom meeting would be recorded, and they could leave the online meeting room or request that the recording be stopped at any time without giving any reason. Each participant was interviewed individually once, in an interview session lasting 60–80 minutes, and no teacher left the conversation before it finished. One of the interviews was excluded from the analysis because of connection issues.

4.4 Data Analysis

The data analysis process started by transcribing comments from the recorded interviews into written text, following suggestions by Braun & Clarke (2006), so familiarity with the data was acquired (step 1). All transcriptions were then imported into Taguette software (Rampin & Rampin, 2021), which was used for coding the transcripts. Taguette was selected because it is free, can be used offline, and is easy to learn. The coding followed a bottom-up thematic approach, striving to ensure that the codes were strongly related to the data with no attempts to fit them into a predetermined coding frame (Braun & Clarke, 2006).

The coding was started by reviewing all teachers' responses. The teachers' responses then were categorised into two groups corresponding to each research question: responses relevant to RQ1 were aggregated under one heading, and those relevant to RQ2 were compiled under another heading. Statements addressing specific components of CT were grouped together under subheadings of the RQ1 category. For instance, all teachers' statements concerning abstraction were initially grouped under the subheading 'Abstraction' and those relate to generalisation were grouped under the heading 'Generalisation'. The coding process for RQ1 was then conducted within each subheading, corresponding to each CT component. Coding for RQ2 was performed without subheadings.

In the first step of the coding process, the main idea of each statement of the teachers was encapsulated by a code consisting of a short phrase. This resulted in more than 230 codes under the headings and subheadings. The analysis continued by looking for themes embedded in each group of codes associated with the subheadings for responses to RQ1 (CT and its six components). The number of themes associated with the components ranged from one to three. Additionally, the statements related to RQ2 initially resulted in three themes. Tables 2 and 3 show illustrative excerpts and codes together with themes that emerged regarding RQ1 and RQ2, respectively.

The analysis was followed by reviewing the themes, then defining and naming them. In the final step one theme related to RQ2 was removed due to the lack of supporting data and consistent explanation, leaving only two themes associated with RQ2. Further details regarding this decision are presented in the Results section (Section 5.2).

Table 2. Examples of excerpts, codes, and themes that emerged related to RQ1

Subheading	Excerpts	Code	Emerged themes
Computational thinking	"An ability to solve problems with certain steps. [It involves] breaking [the problem] into several small parts, then directing [them] to a systematic solution" (Heli)	CT is solving problems by breaking them into smaller parts	CT is problem solving
	"Thinking like a computer. [Since] the language that a computer reads is the language of algorithms, [to simplify it], we [can] read it as reading patterns" (Erick)	CT is thinking like a computer	CT is a particular way of thinking
	"How we can solve a problem. So, we think about how the problem can be solved with our way of thinking" (Paul)	CT is solving problems with our way of thinking	CT is a way of thinking for problem solving
Abstraction	"A word problem related to absolute value; this means that there must be something to pay attention to about the numbers, and then there is something we can ignore" (Robert)	solving word problems related to absolute value	abstraction relates to solving word problems
	"[it] means that we cannot see it using the naked eye" (Andi)	something invisible with naked eye	the abstract nature of mathematical objects
	"When we use mathematical symbols, [students] understand less, but when we use everyday language, [they] sometimes understand better" (Hanna)	teaching strategy to make it easier for students to learn	teaching practices that could ease students' mathematics learning
Algorithmic thinking	"Calculus clearly has an algorithm, such as the derivative [of] $y = a \cdot x^n$, then the derivative must be $a \cdot n \cdot x^{n-1}$. [...] the algorithm is	power rule for differentiation in calculus	step-by-step procedures for solving mathematics tasks

Subheading	Excerpts	Code	Emerged themes
	like that; the students just use that algorithm” (Erick)		
	“Write down all the steps, what is known […], what is being asked, this formula that might be used, […] put what is known into the formula, finally get the solution” (Hanna)	steps of solving a task are to write down what is known and what the task asks for	
Automation	“We can just automate it [drawing a sine function] with GeoGebra” (Paul)	automating drawing process with GeoGebra	using digital tools for solving mathematics tasks
	“The application […] when we scan [math tasks] using a cell phone camera, the answer will immediately come up” (Felix)	use an app to scan tasks and get instant solutions	
	“Automation means the system runs automatically. When, for example, we are given [something], say a function like, function $x = 1$, [then] automatically x [is] replaced with one everywhere” (Andi)	automatic replacement of the value of x in a function	automatic process in mathematics
Debugging	“It usually relates to functions. […] to ensure the solution is correct, we usually check again, substituting the solution back into the function’s formula” (Heli)	substituting a solution back into the function to check that it is correct	checking the correctness of solutions of math tasks
	“[When] the students are given a test, then there is an analysis, then [it is] evaluated, there are [students] who score less [and] how many [students] who score more [than the minimum score]” (Felix)	analysis and evaluation of students’ test results	evaluation of students’ learning
Decomposition	“We will break down it [a complex geometrical shape], from the triangle first, then the rectangle” (Andi)	breaking down a complex geometrical object	breaking down mathematics tasks into smaller parts
	“Before we solve this problem, pay attention to what we need to find out first, for example […] write down what you need to know first” (Robert)	breaking down a problem into what is known and what is unknown	
Generalisation	“Multiplying by 11 […] the quick formula is, separate it, the middle is the sum. I think that’s a pattern too” (Robert)	pattern in multiplying by 11	recognising patterns and similarities in mathematics

Subheading	Excerpts	Code	Emerged themes
	"Generalisation, in my opinion, [is] already very well learned by [students] when they study number patterns. [...] Later in lower elementary school, [they continue learning generalisation in] sequences and series" (Alex)	students learn generalisation in the topic of number patterns, sequences, and series	
	"If [the task is] like this, it means the pattern is a substitution, this means that you will use integral by substitution" (Erick)	recognising task patterns as a strategy for solving integral tasks	

Table 3. Examples of excerpts, codes, and themes that emerged related to RQ2

Excerpts	Code	Emerged themes
"[Integrating CT] is very possible. It depends on the teachers, on who wants to organise it" (Alex)	feasibility of CT integration depends on the teachers	teachers play essential roles in CT integration
"Teachers need to understand CT, then after that teachers also need to be trained" (Robert)	teachers need to understand CT, and the need of training	
"Teachers must understand the step-by-step [process] and its influence on the way of delivering [the lessons] to students" (Heli)	teachers' understanding of CT influences the design of the lessons	
"With problem-based learning, we actually guide students to teach them how to find solutions to a problem. From there, of course, they have started to think in CT" (Paul)	problem-based learning may promote CT	certain practices and tasks could be used to introduce CT
"[tasks that follow the national assessment framework] can definitely be used [to introduce CT]. Because [...] from the existing reading, [students] also have to [analyse] first, maybe there is decomposition" (Alex)	tasks that follow the national assessment framework may promote CT	

5 Results

The results are presented in two sub-sections. The first presents findings related to RQ1, i.e., the teachers' perceptions of CT and connections of its components to mathematics and the second presents their views on the potential integration of CT in teaching and learning mathematics.

5.1 Mathematics teachers' perceptions of computational thinking and the relationship of its components to mathematics

RQ1, "*How do mathematics teachers perceive CT, and how do they associate its components with mathematics?*" was addressed through an open-ended question, "What do you know about CT?". The responses indicate that three teachers viewed CT primarily as problem-solving, emphasising decomposition and pattern recognition. Another three teachers regarded CT as a specific way of thinking, aligning it with recognising patterns and computer-like thought processes. Two others combined both views, describing CT as "thinking like a computer, which is used to solve problems".

The teachers' perceptions of CT were further investigated by asking questions related to each CT component (abstraction, algorithmic thinking, automation, decomposition, debugging, generalisation). While CT was new to them, the teachers indicated that it was not entirely unfamiliar. They recognised that they had already been practising CT components without knowing them as such. Paul, for instance, stated that "even in mathematics itself, … before [CT] was proposed, CT actually already existed". Hanna also expressed the same view, that CT "is not actually new". This familiarity with CT emerged as a consistent theme throughout the interviews. The following subsections present how teachers connected CT components with mathematics.

5.1.1 Abstraction

During the interviews the teachers connected abstraction to solving word problems, the abstract nature of mathematical objects and teaching practices that could ease students' mathematics learning.

Two teachers stated that abstraction is involved in converting word problems into mathematical notions, and hence solving them, by "raising attention to the numbers". A task that focuses on solving equations directly does not require abstraction, but a task in a narrative form requires abstraction and representation of the problem in mathematical expressions before it can be worked on.

Four teachers associated abstraction with the abstract nature of mathematical objects, particularly in the need to think abstractly—imagining something that is not visible—and the transition between concrete and abstract representations in mathematics. One teacher stated that abstraction is thinking of an invisible object and that abstraction happens when students use their imagination to think about an abstract object. Another teacher drew a similar connection, using the imagination of an invisible geometric object to exemplify abstraction. Furthermore, two teachers linked abstraction to the differences between mathematics at the primary and secondary levels. One of them emphasised the transition from concrete mathematics in primary school to a more abstract level in lower secondary school, while the other stated that the introduction of mathematical symbols (as an example of abstraction) is more common at the early levels.

Two teachers linked abstraction to practices that could ease students' mathematics learning after reading the definition of abstraction, which includes the phrase: "… choosing the right detail to hide so that the problem becomes easier, without losing anything that is important".

5.1.2 Algorithmic thinking

No teachers alluded to algorithmic thinking in responses to the open question about CT. All their answers regarding algorithmic thinking came after they were provided with a definition and an example of it. The analysis revealed that the teachers connected algorithmic thinking to step-by-step procedures to solve mathematics tasks.

Two teachers linked algorithmic thinking to the application of stepwise methods in mathematical operations. Examples given by the teachers included step-by-step approaches to long division and algebraic multiplication. Erick specifically linked algorithmic thinking with the power rule for differentiation in calculus. He gave an example that the differentiation of $y = a \cdot x^n$ is $y' = a \cdot n \cdot x^{n-1}$ and stated, "the algorithm is like that, so students just use the algorithm". Robert associated algorithmic thinking with structured steps for solving complex tasks beyond simple calculation, such as integral with u-substitution. He explained, "… for example, to solve a substitution integral, the first step is to determine the analogy (the substitution variable), the second is to substitute this, and so on." Robert also provided an example from linear programming, where solving a task requires creating a table, developing a mathematical model, and identifying of intersection points. Furthermore, another teacher linked algorithmic thinking with the process of determining known and unknown parts of tasks.

5.1.3 Automation

Automation was not mentioned by any teachers during responses to the open question about CT, but related responses came after the teachers were shown the definition of automation. Most participants (seven teachers) related it to the use of digital tools, including computers, calculators, and software like Microsoft Excel and GeoGebra, for solving mathematics tasks. They said that Microsoft Excel helps to avoid the need for manual calculation, and GeoGebra can automate the process of drawing graphs. Similarly, a teacher stated that calculators can help students to perform mathematical computations, and two connected automation to applications (such as Photomath) that can generate instant solutions for mathematical tasks.

Furthermore, one teacher did not mention any digital tools when talking about automation. Instead, he explained that automation relates to automatic processes in mathematics, using the example of the automatic change in the value of a function when a value is assigned to x.

5.1.4 Debugging

Similar to algorithmic thinking and automation, no teachers mentioned debugging when asked what they knew about CT but responded after they were presented the definition of debugging by Bocconi et al. (2016) and an example of debugging a formula in Excel.

While the participating teachers were still unsure about the connections between debugging and mathematics, most (seven teachers) associated it with checking the correctness of solutions to mathematics tasks. This includes verifying solutions during task solving, which can be done by either teachers or students. Verification is required to either ensure the correctness of an answer or to identify mistakes if the answer is incorrect. For instance, Alex said that he sometimes checked students' solutions and if they were not correct, he would ask them to review each step to ensure all the steps were correct. Paul provided an example of verifying solutions for a system of three-variable algebraic equations by substituting the solution back into each equation to check if it satisfies all three equations.

Furthermore, one teacher seemed to focus on the term 'evaluation' when reading the definition of debugging and linked it to evaluating students' learning results through a test.

5.1.5 Decomposition

Three teachers mentioned decomposition during the open question about CT and others made relevant responses after being shown the definition and example of decomposition. The teachers associated decomposition with breaking down mathematics tasks into smaller components.

The teachers mentioned various examples of decomposition in mathematics, such as breaking down geometric objects into smaller parts. One teacher specifically mentioned function composition as an example of decomposition, considering that it helps students to learn that functions may be compound and consist of several functions. Decomposing mathematics tasks also includes breaking down the solving process into smaller operations. For example, drawing quadratic functions can be decomposed into finding some intersection points and factorisation, and solving a word problem may be helped by identifying parts that contain known information and parts that need to be solved. One teacher linked decomposition to a taxonomy by Bloom (1956), suggesting that the need for decomposition to solve a mathematics task depends on the level of thinking required. There is little need for decomposition to solve a C1 task, which only requires the solver to recall facts and basic concepts. However, tasks demanding higher order levels of thinking, such as producing original work or justifying a decision (C5 and C6 tasks, according to the taxonomy) would encourage students to perform decomposition.

5.1.6 Generalisation

Four teachers mentioned generalisation during the open question about CT, while the rest shared their views on generalisation after being presented with the description of CT by Bocconi et al. (2016). The teachers connected generalisation to the recognition of patterns and similarities in mathematics. Their responses regarding generalisation included topics in mathematics that encourage students to practise generalisation, examples of generalisation in mathematics, and its use as a strategy for solving mathematics tasks.

Sequences and series were the most frequently raised examples of generalisation in mathematics (mentioned by four teachers). Examples of generalisation in mathematics include the supplementary nature of opposite angles of a cyclic quadrilateral, and that when any two-digit number is multiplied by 11 the first and last digits remain unchanged, while the middle digit is the sum of the original numbers' digits (and if the sum exceeds 9, carry over the tens digit to the first digit). Furthermore, generalisation was also viewed as a strategy for solving mathematics tasks. One teacher associated the term 'finding pattern' with recognising the characteristics of tasks in integral topics. When one has recognised the pattern of a task in integral, one can decide whether the task needs to be solved using integration by substitution or integration by parts. Generalisation was also associated with teachers asking students, "Have you ever solved a problem like that?". This question aimed to encourage them to see similarities between new information and previously learned topics.

5.2 Mathematics teachers' view of potential computational thinking integration into teaching and learning mathematics

RQ2 concerns the mathematics teachers' views of CT's possible integration into teaching and learning mathematics. One of the emerging themes related to RQ2 was that the teachers held a favourable view of CT integration. However, their explanations for this positive perception were either absent or highly varied. Some teachers simply expressed agreement with CT integration without providing specific reasons. Due to this lack of supporting data and consistent explanation, the theme was removed during the final step of data analysis. Nevertheless, two themes emerged regarding their views of CT's integration into mathematics. One is recognition that teachers play an essential role when CT should be integrated, and the other is recognition of practices and tasks that could be used to expose students to CT.

5.2.1 The essential role of teachers in integrating computational thinking into mathematics

The teachers raised concerns about the importance of teachers when CT is integrated into mathematics. They mentioned that mathematics teachers should have adequate knowledge of CT before its integration becomes mandatory to allow them to identify appropriate topics for introducing specific CT components and designing suitably tailored activities. In addition to recognising the importance of adequate knowledge, the teachers mentioned the need for relevant training. One referred to the connections between CT and some aspects of mathematics, and stated that although mathematics already has some CT components, teachers' training is still required to "convey [to the teachers] that what [they] teach is part of computational thinking."

Furthermore, two teachers emphasised the importance of teachers' willingness to improve their competence. One of them stated that change and improvement would only occur if teachers were willing to learn and be open-minded about new things. Another reflected on his experience and noted that in-service teacher training sessions were frequently attended by the same group of teachers. He later questioned why other teachers did not participate in activities that could improve their professional competence.

5.2.2 Some practices and tasks could be used to introduce computational thinking in mathematics instruction

During the interviews the teachers were asked about the feasibility of integrating CT into mathematics and were invited to suggest ways of doing so. All teachers said that integrating CT into mathematics is feasible.

The teachers mentioned some practices in teaching mathematics that could be useful for exposing students to some CT components and tasks that could potentially foster related abilities. For example, Paul suggested that problem-based learning may encourage students to practice some CT components as it requires students to find solutions for problems, and Erick, reflecting on his experience of using Microsoft Excel in teaching statistics, said that Excel could help students to practice automation. Two teachers stated that teaching mathematics through STEM projects may engage students in practising some CT components, but without further explanation of how this could occur. Furthermore, Alex suggested that giving students worksheets that provide structured guidance to help them solve tasks could also foster the development of CT abilities, e.g., by encouraging them to practice CT components such as decomposition and generalisation. He noted that such worksheets "... make it easier for students to indirectly follow the steps of CT". However, CT inclusion through established classroom activities may require additional instructional time and increase the cognitive demands on students. One teacher anticipated that incorporating CT would require extending certain lessons, thus increasing the time needed for classroom instructions. He further explained that the students would need to make extra efforts, as they would be learning both mathematics and CT simultaneously.

The participants also mentioned characteristics of tasks that may expose students to CT, including word problems and tasks that encourage students to engage in higher-order thinking. Moreover, four teachers suggested that tasks aligned with the new Minimum Competency Assessment framework (Indonesian: Asesmen Kompetensi Minimum, hereafter AKM)[2] can potentially promote students' practice of CT. According to Hanna, tasks aligned with the AKM framework do not only require simple calculation but also reading a text, gathering relevant information and applying it to find correct solutions.

---

[2] Minimum Competency Assessment (*Asesmen Kompetensi Minimum*, AKM) is a part of the new national assessment and serves to measure Indonesian students' literacy (reading comprehension) and numeracy (mathematical skills). Tasks for AKM numeracy are intended to evaluate students' mathematical reasoning, application, and problem-solving, with an emphasis on reasoning skills rather than rote learning (Adiputri, 2023; Pusat Asesmen Pendidikan, 2023).

This type of task, Hanna suggested, could promote students' application of CT in mathematics.

5.3 Summary of the results

In summary, the participating teachers perceived CT as problem-solving and the underlying way of thinking. They also associated CT components with certain parts of mathematics such as linking abstraction to converting word problems into mathematical notation and algorithmic thinking to following procedures to solve mathematics tasks.

Furthermore, the teachers had positive views of CT integration into mathematics education, and recognised that teachers had essential roles in the process. They highlighted the need for training that would equip teachers with relevant knowledge, and identified some practices and tasks that could serve as entry points for CT's introduction.

6 Discussion

In this section the teachers' perceptions of CT are discussed, particularly their perceptions of CT as problem-solving and the factors that may influence the teachers' connections between CT and mathematics. Implications of the findings are also explored, particularly concerning the design of professional development required to support mathematics teachers. Finally, the limitations are discussed along with potential directions for future studies.

6.1 Teachers' perceptions of computational thinking and the potential factors that influence these perceptions

Previous literature shows that teachers have varied perceptions of CT. For example, it has been seen as a tool to foster motivation in teaching and learning (Humble & Mozelius, 2023) and as an add-on to existing curricula (Nordby et al., 2022). The teachers who participated in this study viewed CT as problem-solving and associated thinking processes, in accordance with reported perceptions of other teachers (e.g., Rich et al., 2019) and experts (Kallia et al., 2021).

However, a deeper examination of the teachers' connections between each of the CT components and mathematics indicates that the teachers' perceptions of CT as problem-solving in mathematics are sometimes vague. Several teachers' statements about problem-solving seem to refer to solving mathematics tasks that require higher-order levels of thinking (see Bloom, 1956) and/or align with emphases in the national assessment framework on promoting abilities to solve non-routine tasks and move beyond procedural fluency (MoEC, 2020a). In contrast, some of the teachers' statements seem to refer to solving mathematics tasks that require little or no reflection on underlying mathematical ideas. For example, some teachers associated algorithmic thinking with merely following procedures and connected automation to using applications to scan mathematics tasks for instant answers. However, solving mathematics tasks without considering the mathematical ideas could lead to rote learning, as students may simply memorise the procedures with no attempt to understand the underlying concepts. This could undermine the original purpose of integrating CT into the educational system, as Wing (2006) stressed that CT is not rote learning but is intended to equip students with fundamental skills.

The findings also show that the teachers associated CT components with specific parts of mathematics, indicating that they regarded it as being already covered by aspects of their current classroom activities (cf. Nordby et al., 2022). The association may be linked to several factors, such as the established use of certain CT terms in mathematics. Generalisation was associated with the topic of sequence, as Indonesian students often learn to identify patterns and express sequences in generalised n-th term form. Decomposition was linked to composite functions, as this topic includes both combining two or more functions into single functions and breaking down composite functions into constituent functions.

In addition, as previously mentioned, the teachers regarded CT as problem-solving, and this perception may have influenced how they connected CT components to mathematics. For instance, some mentioned the phrases "what is known" and "what is asked for" when talking about algorithmic thinking and decomposition. These phrases are typical in Indonesian school mathematics education when teaching students how to solve word problems and might be at least partly inspired by Polya's (2004) strategy for solving a problem: identifying its known and unknown parts. Furthermore, some teachers' connections did not align with the definition of CT used in this study. For instance, abstraction was associated with efforts to ease students' learning, and automation was linked to automatic change of the value of a function. In addition, they were also unsure how to illustrate debugging in mathematics and the examples they did provide (e.g., error analysis, rechecking work) reflect traditional mathematical practices. A potential explanation for such misalignment and uncertainty is that the teachers did not have clear ideas about specific CT components. Consequently, when certain definitions of CT components were shown to them, the teachers focused on particular aspects of the definitions and drew connections based on those aspects rather than the complete definitions.

6.2 Professional development to support computational thinking integration: design considerations and teachers' willingness

Professional development is widely recognised as essential for supporting mathematics teachers to integrate CT (Liu et al., 2024) and its effective design requires careful attention to teachers' perceptions. Acknowledging teachers' perceptions would enable designers of professional development to know the starting point of the programme and address potential misconceptions. The finding of the study, that teachers' perceptions of abstraction, automation, and debugging do not fully correspond to CT definitions, can inform the designers to pay attention to these three aspects. Furthermore, as problem-solving is commonly recognised as a key characteristic of CT in mathematics (cf., Huang et al., 2021; Kallia et al., 2021), a development programme could use it as the basis for its activities. This study found that teachers' perception of CT as problem-solving may include solving tasks that encourage rote learning which contrasts with CT's original aim of equipping students with fundamental skills. Thus, selecting tasks for integration is crucial for optimising the benefits of integrating CT into mathematics education. If professional development emphasises problem-solving as a bridge to introduce CT, it should encourage teachers to utilise tasks that promote students' conceptual understanding and avoid those that make students rely on memorised procedures.

Moreover, recognising the educational context, including the current curriculum and longstanding teaching practices, is crucial to ensure that the designed professional development is meaningful and relevant for teachers. As discussed earlier, the teachers connected CT components with specific topics from the Indonesian school mathematics curriculum, and they identified examples of established practices and characteristics of tasks that could serve as starting points for CT integration. These findings suggest that supporting mathematics teachers to integrate CT should focus on enhancing and adapting existing teaching practices rather than enforcing significant changes. Professional development that encourages teachers to reframe their existing practices to highlight CT components would elevate their confidence, helping them to see CT integration as achievable and meaningful. However, when embedding CT into existing practices, it is essential to consider several caveats related to teachers' concerns. These include concerns of additional time required to prepare and conduct CT-embedded lessons and increased cognitive demands on students, who need to learn two things simultaneously.

Alongside design considerations, professional development supporting mathematics teachers integrate CT can take various forms. These include workshops that encourage the development of CT-infused lesson materials, allowing teachers to design and tailor lessons that incorporate CT to specific topics and their own instructional needs. Another potential form of professional development involves classroom-focused activities, such as peer observation of CT-aligned lessons, which create opportunities for teachers to learn from each other and reflect on their instructional practices.

Furthermore, the findings indicate that the participating teachers had a positive outlook on CT integration, and recognised their essential role in the process, emphasising their needs for adequate knowledge, in accordance with previous studies (Humble & Mozelius, 2023; Nordby et al., 2022; Reichert et al., 2020; Vinnervik, 2020). Although the sample in this study may not be fully representative, these findings offer valuable insights that mathematics teachers may be willing to learn about CT and actively participate in professional development. Teachers' motivation to engage in professional development is crucial for its success (Guskey, 2002), and their willingness could encourage governments and regulators to provide professional development to support CT's incorporation into mathematics education. On top of that, the teachers' willingness also shows that there are promising opportunities for researchers to further study CT's integration into mathematics, particularly through collaboration with mathematics teachers.

6.3 Limitations and future directions

This study contributes to the existing body of knowledge on mathematics teachers' perceptions of CT and its integration into mathematics. It offers valuable insights for researchers and curriculum designers on how to prepare support for mathematics teachers involved in the inclusion of CT in teaching and learning mathematics. While the aim was to obtain rich insights into mathematics teachers' views of CT integration in mathematics rather than generalisable results, the relatively small sample size should be taken into account when interpreting the findings, as a larger sample may have provided more diverse teachers' perspectives.

Two themes emerged regarding mathematics teachers' views on integrating CT into mathematics: the role of teachers, and the consideration of established teaching practices and tasks. Building on these insights, subsequent empirical studies can explore how each factor influences the integration and how their interaction supports effective CT-infused mathematics lessons. Additionally, given its focus on teachers' views on CT, this study does not extend to how teachers' perspectives may impact students' learning; future investigations through classroom observations, particularly in under-resourced schools, would offer insights into practical implications of teachers' perceptions.

It is also worth noting that the present study involved a group of teachers who had participated in government programmes intended to introduce CT. These teachers may have attended the programmes due to their interest in CT integration, which may have contributed to their positive outlook on it. Thus, different results may be obtained from studies involving other groups of teachers. In addition, although the participants varied in teaching experience, no consistent differences were observed between early-career and more experienced teachers in their views on CT and its integration into mathematics education. This opens the door for future research to systematically investigate how the variations in teacher experience may influence teachers' views on CT.

The present study lacked access to the structure and materials of the government programmes, which may have influenced the teachers' view of CT. Further studies would benefit from access to the programme materials and consideration of the materials during analysis to better understand whether the programmes' structure and content contribute to shaping teachers' views on CT and its integration into mathematics. Moreover, the definition of CT by Bocconi et al. (2016) was used to structure the interviews in this study, and as previously discussed, may have influenced some of the teachers' responses. Future studies could explore the use of different descriptions of CT and their impacts on different teachers' perspectives. Finally, it should be noted that the inclusion of CT into mathematics classrooms in Indonesia is still at an early stage and it would be valuable to conduct subsequent research comparing CT integration with that of countries where the integration is more mature.

Research ethics

Institutional review board statement

The study was conducted in accordance with the ethical guidelines recommended by The Finnish National Board on Research Integrity TENK. The research involved adult participants (teachers), thus a review by institutional ethics board was not required. Participation was voluntarily and informed consent was obtained from all participants.

Informed consent statement

Informed consent was obtained from all research participants.

Data availability

The data that support the findings of this study are available from the author upon reasonable request.

Acknowledgements

The author expresses gratitude to Ewa Bergqvist and Björn Palmberg for their guidance and insightful comments that enriched this article. Thanks also to Carina Granberg for valuable suggestions on the initial draft.

Conflicts of interest

The author declares no conflicts of interest.

Appendix A Interview guidelines

The interview consists of three sessions: (1) an open-ended discussion on computational thinking (CT), (2) an exploration of the six CT components as described by Bocconi et al. (2016), and (3) discussions on the potential integration of CT into mathematics instruction.

Session 1: An open question about Computational Thinking

> Opening Question: What do you know about computational thinking?
>
> Follow-up Questions: If the participant mentions any of the six CT components (abstraction, algorithmic thinking, automation, debugging, decomposition, or generalisation), the interviewer notes the terms and follows up with tailored questions. For example, if a participant mentions "abstraction," the follow-up questions include:
>
> 1.1. What do you mean by abstraction? 1.2. What mathematical topics do you think relate to abstraction? 1.3. How do you teach these topics? 1.4. Do your students use abstraction when they do mathematics? Can you give examples? 1.5. What suggestions would you give to teachers who teach topics that involve abstraction?
>
> (The same questions apply to all terms mentioned by participants)

Session 2: Discussion on CT Components

The researcher presents the definition of CT by Bocconi et al. (2016) (not repeated here, see Section 2.2 of this article).

The interviewer asks about components not mentioned in Session 1. For example, if a participant has not discussed debugging, the following questions are asked:

2.1. Do you know what is debugging? 2.2. What mathematical topics do you think relate to debugging? 2.3. How do you teach these topics? 2.4. Do your students use debugging when they do mathematics? Can you give examples? 2.5. What suggestions would you give to teachers who teach topics that involve debugging?

If the participant does not know a component (e.g., debugging), the interviewer provides its definition (see Section 2.2 of this article) and examples (see below) before continuing with questions 2.2 to 2.5.

Examples of CT components:

- *Abstraction:* (1) Maps contain important information about roads while omitting unnecessary details like trees, (2) Numbers are abstractions of quantity (e.g., 7 represents seven apples, oranges, or chairs), (3) Converting a word problem into a mathematical model requires selecting essential aspects and ignoring irrelevant details.
- *Decomposition.* (1) Solving 25*3 by decomposing it into (20+5)*3, (2) Finding the area of a trapezium by splitting it into a rectangle and a triangle.
- *Generalisation:* (1) Recognising that the sum of an even and odd integer is always odd, (2) Generalising exponent rules: $a^m * a^n = a^{m+n}$.
- *Algorithmic Thinking:* (1) The multiplication algorithm for two-digit numbers follows defined steps, (2) Finding the greatest common factor (GCF) using factor trees.
- *Debugging:* (1) Checking an Excel formula for incorrect syntax or logical errors when calculating student grades.
- *Automation:* (1) Automation of school bell, (2) Algorithm of multiplication can be automated by building a calculator program, for example, in Phyton.

Session 3: Computational Thinking in Mathematics Instruction

3.1. What do you think would happen if the government changed the curriculum to include CT as a focus in mathematics lessons? 3.2. Is it possible to design activities that allow students to learn mathematics while developing computational thinking? Why? 3.3. Would you face any challenges if you considered CT aspects as additional learning goals? Why? 3.4. What factors might make integrating CT into mathematics difficult or easy for students? 3.5. Do you think integrating CT aspects into your teaching benefits you or your students? Why? 3.6. Are there specific tasks that could help math teachers integrate CT?

References

Anderson, N. D. (2016). A call for computational thinking in undergraduate psychology. *Psychology Learning & Teaching*, *15*(3), 226–234. https:/doi.org/10.1177/1475725716659252

Bailey, D. H., & Borwein, J. M. (2011). Exploratory experimentation and computation. *Notices of the American Mathematical Society*, *58*. https://doi.org/10.1177/1475725716659252

Baldwin, D., Walker, H. M., & Henderson, P. B. (2013). The roles of mathematics in computer science. *ACM Inroads*, *4*(4), 74–80. https://doi.org/10.1145/2537753.2537777

Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning & Leading with Technology*, *38*(6), 20–23.

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

Bloom, B. S. (1956). *Taxonomy of educational objectives: The classification of educational goals*. Longman Group.

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). Developing computational thinking in compulsory education. *European Commission, JRC Science for Policy Report*, *68*.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*(2), 77–101. https://doi.org/10.1191/1478088706qp063oa

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 Annual Meeting of the American Educational Research Association, Vancouver, Canada*, *1*, 25. https://web.media.mit.edu/~kbrennan/files/Brennan_Resnick_AERA2012_CT.pdf

Cheng, Z.-J. (2012). Teaching young children decomposition strategies to solve addition problems: An experimental study. *The Journal of Mathematical Behavior*, *31*(1), 29–47. https://doi.org/10.1016/j.jmathb.2011.09.002

Cohen, L., Manion, L., & Morrison, K. (2018). *Research methods in education*. Routledge.

Computer Science Teachers Association (CSTA). (n.d.). *The CSTA K–12 computer science standards*. CSTA. Retrieved 11 June 2025, from https://csteachers.org/k12standards/

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). *Computational thinking-A guide for teachers*. Computing At School.

Dagienė, V., Jevsikova, T., Stupurienė, G., & Juškevičienė, A. (2022). Teaching computational thinking in primary schools: Worldwide trends and teachers' attitudes. *Computer Science and Information Systems*, *19*(1), 1–24. https://doi.org/10.2298/CSIS201215033D

Denning, P. J. (2017). Remaining trouble spots with computational thinking. *Communications of the ACM*, *60*(6), 33–39. https://doi.org/10.1145/2998438

Dörfler, W. (1991). Forms and means of generalization in mathematics. In A. J. Bishop, S. Mellin-Olsen, & J. Van Dormolen (Eds.), *Mathematical knowledge: Its growth through teaching* (Vol. 10, pp. 61–85). Springer Netherlands. https://doi.org/10.1007/978-94-017-2195-0_4

Elicer, R., Tamborg, A. L., Bråting, K., & Kilhamn, C. (2023). Comparing the integration of programming and computational thinking into Danish and Swedish elementary mathematics curriculum resources. *LUMAT: International Journal on Math, Science and Technology Education*, *11*(3). https://doi.org/10.31129/LUMAT.11.3.1940

Guskey, T. R. (2002). Professional development and teacher change. *Teachers and Teaching*, *8*(3), 381–391. https://doi.org/10.1080/135406002100000512

Hsu, Y.-C., Irie, N. R., & Ching, Y.-H. (2019). Computational thinking educational policy initiatives (CTEPI) across the globe. *TechTrends*, *63*(3), 260–270. https://doi.org/10.1007/s11528-019-00384-4

Huang, W., Chan, S. W., & Looi, C. K. (2021). Frame shifting as a challenge to integrating computational thinking in secondary mathematics education. *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, 390–396. https://doi.org/10.1145/3408877.3432400

Humble, N., & Mozelius, P. (2023). Grades 7–12 teachers' perception of computational thinking for mathematics and technology. *Frontiers in Education*, *8*, 956618. https://doi.org/10.3389/feduc.2023.956618

International Society for Technology in Education (ISTE). (n.d.). *ISTE computational thinking competencies guide*. ISTE. Retrieved 11 June 2025, from https://iste.org/standards/computational-thinking-competencies

Kallia, M., van Borkulo, S. P., Drijvers, P., Barendsen, E., & Tolboom, J. (2021). Characterising computational thinking in mathematics education: A literature-informed Delphi study. *Research in Mathematics Education*, *23*(2), 159–187. https://doi.org/10.1080/14794802.2020.1852104

Kaput, J. J. (2017). What is algebra? What is algebraic reasoning? In J. J. Kaput, D. W. Carraher, & M. L. Blanton (Eds.), *Algebra in the early grades* (1st ed., pp. 5–18). Routledge. https://doi.org/10.4324/9781315097435-2

Knuth, D. E. (1985). Algorithmic thinking and mathematical thinking. *The American Mathematical Monthly*, *92*(3), 170–181. https://doi.org/10.1080/00029890.1985.11971572

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

Ling, U. L., Saibin, T. C., Labadin, J., & Aziz, N. A. (2017). Preliminary investigation: Teachers' perception on computational thinking concepts. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, *9*(2–9), 23–29.

Liu, Z., Gearty, Z., Richard, E., Orrill, C. H., Kayumova, S., & Balasubramanian, R. (2024). Bringing computational thinking into classrooms: A systematic review on supporting teachers in integrating computational thinking into K-12 classrooms. *International Journal of STEM Education*, *11*(1), 51. https://doi.org/10.1186/s40594-024-00510-6

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

Marom, S. (2023). Berpikir komputasi di dalam kurikulum merdeka: Analisis pada guru matematika. *Jurnal Matematika Dan Pendidikan Matematika*, *14*(1).

Ministry of Education and Culture of Indonesia [MoEC]. (2019). *Pedoman implementasi muatan/mata pelajaran Informatika Kurikulum 2013*. Ministry of Education and Culture of Republic Indonesia.

Ministry of Education and Culture of Indonesia [MoEC]. (2020a). *AKM dan implikasinya pada pembelajaran*. Ministry of Education and Culture of Republic Indonesia.

Ministry of Education and Culture of Indonesia [MoEC]. (2020b). *Naskah akademik muatan Informatika dalam Kurikulum 2013*. Ministry of Education and Culture of Republic Indonesia.

Mitchelmore, M. C. (2002). The role of abstraction and generalisation in the development of mathematical knowledge. *Proceeding of the 2nd East Asia Regional Conference on Mathematics Education*, 157–167.

Natali, V., Natalia, N., & Nugraheni, C. E. (2023). Indonesian Bebras challenge 2021 exploratory data analysis. *Olympiads in Informatics*, *17*, 65–85. https://doi.org/10.15388/ioi.2023.06

Nordby, S. K., Bjerke, A. H., & Mifsud, L. (2022). Primary mathematics teachers' understanding of computational thinking. *KI - Künstliche Intelligenz*, *36*(1), 35–46. https://doi.org/10.1007/s13218-021-00750-6

Nouri, J., Zhang, L., Mannila, L., & Norén, E. (2020). Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, *11*(1), 1–17. https://doi.org/10.1080/20004508.2019.1627844

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Polya, G. (2004). *How to solve it: A new aspect of mathematical method* (Issue 246). Princeton university press.

Prahmana, R. C. I., Kusaka, S., Peni, N. R. N., Endo, H., Azhari, A., & Tanikawa, K. (2024). Cross-Cultural insights on computational thinking in geometry: Indonesian and Japanese students' perspectives. *Journal on Mathematics Education*, *15*(2), 613–638.

Rampin, R., & Rampin, V. (2021). Taguette: Open-source qualitative data analysis. *Journal of Open Source Software*, *6*(68), 3522. https://doi.org/10.21105/joss.03522

Regulation of the Minister of Education and Culture (MoEC) of the Republic of Indonesia No. 35 of 2018 on Amendments to the Regulation of the Minister of Education and Culture No. 58 of 2014 on the 2013 Curriculum for Junior High Schools (Sekolah Menengah Pertama) and Islamic Junior High Schools (Madrasah Tsanawiyah) (2018).

Regulation of the Minister of Education and Culture (MoEC) of the Republic of Indonesia No. 36 of 2018 on Amendments to the Regulation of the Minister of Education and Culture No. 59 of 2014 on the 2013 Curriculum for Senior High Schools (Sekolah Menengah Atas) and Islamic Senior High Schools (Madrasah Aliyah) (2018).

Regulation of the Minister of Education, Culture, Research, and Technology (MoECRT) of the Republic of Indonesia No. 12 of 2024 on the Curriculum for Early Childhood Education, Basic Education, and Secondary Education (2024).

Reichert, J. T., Couto Barone, D. A., & Kist, M. (2020). Computational thinking in K-12: An analysis with mathematics teachers. *Eurasia Journal of Mathematics, Science and Technology Education*, *16*(6). https://doi.org/10.29333/ejmste/7832

Rich, K. M., & Yadav, A. (2020). Applying levels of abstraction to mathematics word problems. *TechTrends*, *64*, 395–403. https://doi.org/10.1007/s11528-020-00479-3

Rich, K. M., Yadav, A., & Schwarz, C. V. (2019). Computational thinking, mathematics, and science: Elementary teachers' perspectives on integration. *Journal of Technology and Teacher Education*, *27*(2), 165–205.

Schley, D. R., & Fujita, K. (2014). Seeing the math in the story: On how abstraction promotes performance on mathematical word problems. *Social Psychological and Personality Science*, *5*(8), 953–961. https://doi.org/10.1177/1948550614539519

SEAMEO QITEP in Mathematics. (2021, March 8). *SEAQIM Launched New Workshop: STEM & Computational Thinking*. SEAMEO QITEP in Mathematics. Retrieved 11 June 2025, from https://www.qitepinmath.org/en/our-news/seaqim-launched-new-workshop-stem-computational-thinking/

SEAMEO Regional Open Learning Center. (2023). *Technology in education: A case study on Indonesia*. https://doi.org/10.54676/WJMY7427

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142–158. https://doi.org/10.1016/j.edurev.2017.09.003

Stupurienė, G., Lucas, M., & Bem-Haja, P. (2024). Teachers' perceptions of the barriers and drivers for the integration of Informatics in primary education. *Computers & Education*, *208*, 104939. https://doi.org/10.1016/j.compedu.2023.104939

Supatmiwati, D., Suktiningsih, W., Winarti, N. K. S., Muhid, A., & Abdussamad, Z. (2024). Sosialisasi dan pelatihan gerakan PANDAI (Pengajar Era Digital Indonesia) pada guru SDN 8 Sokong Lombok Utara. *Jurnal Mengabdi Dari Hati*, *3*(2), 63–70.

Susilowati, D., Ria, R. R. P., Anggriani, R., Salsabila, S. I., Aditia, R., & Aziz, F. (2025). Peningkatan kompetensi guru STEM dalam pembelajaran informatika berbasis HOTS dan computational thinking guna mendukung implementasi Kurikulum Merdeka di MTsN I Mataram. *Journal of Community Development*, *5*(3), 507–516.

Suwaji, U. T., Guntoro, S. T., & Wiworo, W. (2020). *Kapita selekta matematika SMA: Pembelajaran berorientasi kemampuan berfikir tingkat tinggi berbasis PISA dan TIMSS*. Ministry of Education and Culture of Republic Indonesia.

Trisnapradika, G. A., Pertiwi, A., Prabowo, W. E. A., Setiyanto, N. A., & Sumarjono, C. A. P. (2024). Pelatihan Model Computational Thinking bagi Guru TK dan SD Gaussian Kamil School Semarang. *Abdimasku: Jurnal Pengabdian Masyarakat*, *7*(2), 576–583.

Ukkonen, A., Pajchel, K., & Mifsud, L. (2024). Teachers' understanding of assessing computational thinking. *Computer Science Education*, 1–26.

Vinnervik, P. (2020). Implementing programming in school mathematics and technology: Teachers' intrinsic and extrinsic challenges. *International Journal of Technology and Design Education*. https://doi.org/10.1007/s10798-020-09602-0

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. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33–35. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Wing, J. M. (2011). Research notebook: Computational thinking—What and why? *The Link: The Magazine of Carnegie Mellon University's School of Computer Science*. https://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why

Wonohadidjojo, D. M., Citra, C. C., Tanamal, R., Soekamto, Y. S., & Maryati, I. (2021). Workshop computational thinking untuk guru SD, SMP dan SMA oleh Biro Bebras Universitas Ciputra Surabaya. *Jurnal Leverage, Engagement, Empowerment of Community (LeECOM)*, *3*(2).

Yunianto, W., Bautista, G., Jr., & Lavicza, Z. (2024). Learning numbers, place values, and CT skills with a spreadsheet. *Proceedings of the 17th ERME Topic Conference MEDA 4*, 460.

Yunianto, W., Cahyono, A. N., Prodromou, T., El-Bedewy, S., & Lavicza, Z. (2025). CT integration in STEAM learning: Fostering students' creativity by making Batik stamp pattern. *Science Activities*, *62*(1), 26–52.

Yunianto, W., El-Kasti, H. S., Prahmana, R. C. I., & Lavicza, Z. (2024). A constructionist approach to learning computational thinking in mathematics lessons. *Journal of Information Technology Education: Innovations in Practice*, *23*, 11.

Zazkis, R., & Liljedahl, P. (2002). Generalization of patterns: The tension between algebraic thinking and algebraic notation. *Educational Studies in Mathematics*, *49*, 379–402. https://doi.org/doi.org/10.1023/A:1020291317178