The Impact of Computational Thinking Skills on Mathematics Learning – Systematic Literature Review

Adi Janes Ngingi[1], Ariyadi Wijaya[2], Fitriana Yuli Saptaningtyas[3]
[1,2,3] Mathematics Education, Yogyakarta State University, Indonesia
*adijanes.2025@student.uny.ac.id

Abstract

The demands of 21st-century education require students to possess higher-order thinking skills, with computational thinking (CT) being one of the core competencies. Although CT is crucial in helping students solve mathematical problems structurally, reality shows that students' CT skills remain moderate, and classroom instructions are predominantly procedural or rote-based. This study aims to systematically analyze and comprehensively map the impacts and research trends of integrating computational thinking into mathematics learning across various educational levels. This study employed a Systematic Literature Review (SLR) method guided by the PRISMA framework. Literature searching was executed using the Scopus database, filtering for peer-reviewed journal articles, written in English or Indonesian, available in open-access format, and published between 2020 to 2028. Based on strict inclusion and exclusion criteria, the final selected articles were analyzed in-depth using bibliometric analysis and thematic synthesis. The synthesis reveals that integrating CT into mathematics instructions yields significant positive impacts at elementary, junior, and senior high school levels. The core components of CT (decomposition, pattern recognition, abstraction, and algorithmic thinking) effectively enhance students' problem-solving capacities, logical reasoning, creativity, and learning motivation. Furthermore, the deployment of innovative approaches such as game-based learning, the 5E instructional model, TPACK, STEM, ethnomathematics, and digital tools (GeoGebra, Scratch, ChatGPT) consistently reinforces instructional efficacy. Nonetheless, critical challenges emerge, including variations in students' cognitive prior knowledge, constraints in assessment instruments, and teachers' misconceptions that narrow CT down strictly to computer programming or coding activities. The integration of CT serves as a vital bridge to transform mathematics pedagogy into a more explorative and meaningful experience, highlighting the urgent need for continuous teacher professional development and standardized CT assessment tools

Keywords: computational thinking, mathematics learning

Introduction

The development of science, technology, and social life in the 21st century has brought about major changes in the way humans learn, work, communicate, and solve problems. 21st-century learning demands high-level thinking skills, such as critical, creative, collaborative, communicative, and adaptive thinking, which are the foundation for dealing with the complexities of the modern world (Wing, 2006; Hartawan, 2024). Students no longer have sufficient control over formulas or procedures; they must also be able to analyse problems, choose solution strategies, process information, recognise patterns, formulate logical thinking steps, and re-evaluate the solutions obtained (Mangiri & Prabawanto, 2024).

One of the key skills in 21st-century learning is computational *thinking*. This ability is not only related to computers or programming, but also a way of thinking that helps understand and solve problems in a structured way (Weintrop et al., 2016; Hartawan, 2024). Computational thinking helps students break complex problems into smaller parts, identify patterns in available information, select relevant information, formulate solution steps, and re-examine whether the solution is correct (Hartawan, 2024; Lehmann, 2024).

The components of computational thinking show their relevance to the student's level of thinking. At the low thinking level, students are only asked to remember or imitate procedures, while at the high thinking level, students are required to analyse situations, choose strategies, evaluate processes, and create solutions (Hendriyani, 2020; Yahfizham, 2023). Decomposition, abstraction, pattern recognition, algorithmic thinking, generalisation, debugging, and solution design are complex mental activities that involve high-level thinking and help students build a systematic thinking process (Weintrop et al., 2016; Looi et al., 2024).

The components of computational thinking are closely aligned with the mathematical thinking process. Decomposition helps break down big problems, pattern recognition helps find order, abstraction sorts out important information, and algorithmic compiles resolution steps. Debugging and verification help students re-examine errors (Looi et al., 2024; Hartawan, 2024). In mathematics, activities such as story problems, number patterns, mathematical models, geometry, and data analysis require computational thinking skills (Mangiri & Prabawanto, 2024; Weintrop et al., 2016).

Mathematics is one of the subjects that has the potential to develop students' computational thinking, because it not only teaches numbers and formulas, but also ways of thinking (Wing, 2006; Richardo et al., 2023). The activity of observing patterns, making generalisations, constructing arguments, building models, and solving problems gradually are computational thinking processes that arise naturally when learning mathematics (Hartawan, 2024).

Systematic studies show that the integration of computational thinking in mathematics learning has been widely developed at various levels, especially in elementary and secondary schools (Mangiri & Prabawanto, 2024; Hartawan, 2024). The media used is diverse, ranging from programming platforms and visualisation tools to interactive simulations, educational games, and mathematical software such as GeoGebra and ChatGPT (Urhan & Kocadere, 2024; Wardat et al., 2023).

However, field data show that many students still have difficulty solving mathematical problems that require deep reasoning (Cahdriyana & Richardo, 2020; Hartawan, 2024). Students tend to imitate examples or directly calculate without understanding the problem's structure. When questions are presented contextually, students have difficulty sorting out important information, determining strategies, and devising appropriate completion steps. If learning is still centred on teachers and memorisation, computational thinking skills are difficult to develop optimally (Hendriyani, 2020; Hartawan, 2024).

Research shows that students' computational thinking skills in solving mathematical problems remain in the medium range, especially in the abstraction, algorithmic, and verification components (Lehmann, 2024; Hartawan, 2024). Students with *low prior knowledge* often misinterpret the problem, leading to inappropriate solution steps. This suggests that the development of computational thinking should take into account students' cognitive readiness and prior learning experiences (Ng & Cui, 2021; Hartawan, 2024).

The application of computational thinking helps students understand problems more structurally, recognise patterns, filter information, formulate procedures, and develop debugging skills (Looi et al., 2024; Hartawan, 2024). This improves the quality of mathematical problem-solving because computational thinking works alongside problem-solving strategies (Lehmann, 2024; Ng et al., 2023).

The integration of cultural and technological contexts, such as ethnomathematics, STEM, GeoGebra, and ChatGPT, can increase students' relevance and motivation (Urhan & Kocadere, 2024; Wardat et al., 2023). The use of GeoGebra and ChatGPT supports concept visualisation, algorithmic training, and the development of debugging and abstraction capabilities (Urhan & Kocadere, 2024; Yunanto et al., 2024).

The application of computational thinking faces challenges, including limitations in teacher understanding and in assessment. CT is not just a part of informatics or programming; CT can be developed through computer-free mathematics learning (Weintrop et al., 2016; Hartawan, 2024). CT assessment involves many cognitive aspects: decomposition, abstraction, algorithms, generalisation, evaluation, and debugging (Looi et al., 2024; Hartawan, 2024).

The main problem is that the development of CT in mathematics learning has not been optimal, even though this ability is highly needed in the 21st century (Wing, 2006; Hartawan, 2024). Students still have difficulty understanding problems, sorting out important information, strategising, recognising patterns, and examining answers.

In Supiarmo & Susanti (2021) research, students' computational thinking skills in solving mathematical problems are still low. Most students are only able to show indicators of pattern recognition and decomposition of problems. Meanwhile, the other two aspects such as abstraction have not been able to be shown by students and students' algorithmic thinking skills that have not been written completely and are not systematic.

In line with that by Supiarmo, et.al. (2021), that in solving linear program problems, students still apply general procedures such as using examples, substitution, elimination but only up to the stage of decomposition and pattern recognition, not yet able to demonstrate abstraction skills. In addition, thinking algorithms have not been able to be shown logically and systematically. Marchelin research et.al. (2022) students also still have difficulty in showing computational thinking skills in solving comparative problems.

The influence of students' low computational thinking skills is caused by several factors, including the learning model carried out by teachers has not provided freedom and facilitated for students to develop computational thinking skills (Weintrop, et.al., 2016; Gadanidis, et.al., 2017). Teachers' creativity in innovating learning is also the cause of students' low computational thinking skills (Tedre & Denning, 2016). teachers also often emphasize mathematics learning which requires students to memorize formulas and routine procedures, thus causing students' computational thinking skills to be low (Gadanidis, et.al, 2017; Angeli & Giannakos, 2020).

Computational thinking is important to be applied in mathematics learning, as it helps students think logically, creatively, and reflectively. CT allows students to break down problems, identify patterns, filter information, devise steps, and correct mistakes, thereby preparing them for academic, technological, and real-life challenges (Weintrop et al., 2016; Urhan & Kocadere, 2024; Hartawan, 2024).

Method

In this study, the four stages of article selection based on the PRISMA flow chart (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) is a standardized protocol that outlines the four main steps in systematic article selection: identification, screening, eligibility, and inclusion. It ensures transparency and reproducibility in the article selection process (Mengist et al., 2020). Therefore, this SLR study covered five main aspects for the articles obtained: search strategy, selection criteria, selection process, data collection, and data analysis. Data was obtained from two databases, Web of Science and Scopus because both provide articles with high academic quality that have gone through the peer review process. This database selection was done to maintain the quality and validity of the systematic review results. Then a search process was carried out for SLR using two keywords "Computational Thinking", "Mathematics Learning".

1. Literature Search Strategy

Harzing's Publish or Perish is used as a tool to search for scientific articles relevant to the research topic. In this process, three main databases were used: Google Scholar, Semantic, and Scopus, to ensure that the sources obtained were comprehensive, highly, credible, and accountable. To narrow the scope of the search and make the result more relevant, number of keywords were used, such as computational thinking, crticial thinking, STEM, TPACK, and mathematics.

The range of publications reviewed is limited to the period 2020–2026 to capture the latest developments and research trends in critical thinking assessment in educational contexts. The search strategy is designed to include a variety of terms that represent computational thinking, critical thinking skills, and assessment and measurement approaches.

To maintain the quality and accessibility of sources, searches are limited to *peer-reviewed* journal articles written in English and available in *open-access* format, enabling open reader review and verification.

2. Inclusion and Exclusion Criteria

Articles obtained from the search results are then selected using predetermined inclusion and exclusion criteria. Establishing these criteria ensures that only relevant, up-to-date, and high-quality studies are included in the analysis.

The inclusion criteria include *peer-reviewed* journal articles published between 2020 and 2025 that explicitly address *computational thinking* education and mathematics learning and are available in open access. In contrast, non-journal publications such as conference proceedings, books, book chapters, editorials, and articles with limited access were excluded from the study. Details of inclusion and exclusion criteria are presented in Table 1.

Table 1. Inclusion and Exclusion Criteria

Category	Inclusion Criteria	Exclusion Criteria
Type of Publication	Peer-reviewed journal articles	Non-journal publications (conference, book, website)
Publication Year	2020–2026	Before 2019
Keywords	Computational Thinking and Mathematics learning	Unrelated or missing keywords
Language	English and Indonesian	Non-English articles
Access Type	Open access (Gold/Green/Hybrid)	Restricted-access publications

Screening Process and Study Selection

The initial search yielded 200 articles from the Scopus database. After eliminating duplication and implementing publication-year limits (2020–2026), the number of articles was reduced to 150. Furthermore, the screening process is carried out in stages in accordance with PRISMA guidelines. At the initial screening stage, articles in the form of book chapters (25), books (2), review articles (5), conference proceedings (91), editorials (1), non-English articles (1), irrelevant topics (36), and articles not available in *open access* format (37) were excluded from the analysis. This process resulted in 30 articles that met all inclusion criteria and were deemed worthy of further analysis.

1. Study Quality Assurance and Analysis Procedures

   A total of 30 selected articles were analysed in depth using two complementary analytical approaches: bibliometric analysis and thematic synthesis. Bibliometric analysis was used to map research development, publication trends, computational thinking skills, and research patterns related to critical thinking assessment during the study period. This approach primarily addresses RQ1 on research evolution and trends.

   Furthermore, thematic synthesis was conducted to examine the substantive content of each article by extracting information on the types of computational thinking and mathematics learning instruments, as well as the psychometric characteristics reported, including evidence of validity, reliability, and construct structure. This thematic analysis plays a major role in answering RQ2 and RQ3, particularly regarding the instrument's implementation, methodological strengths and limitations, and future research directions. To ensure the quality of the study, only *peer-reviewed articles* that meet the inclusion criteria are analysed. In addition, the process of extracting and grouping data is carried out systematically based on predetermined analysis categories, thereby reducing the potential for bias and increasing the consistency of interpretation.A summary of the overall process of identification, screening, and study selection is presented in Figure 1 via the PRISMA flowchart.

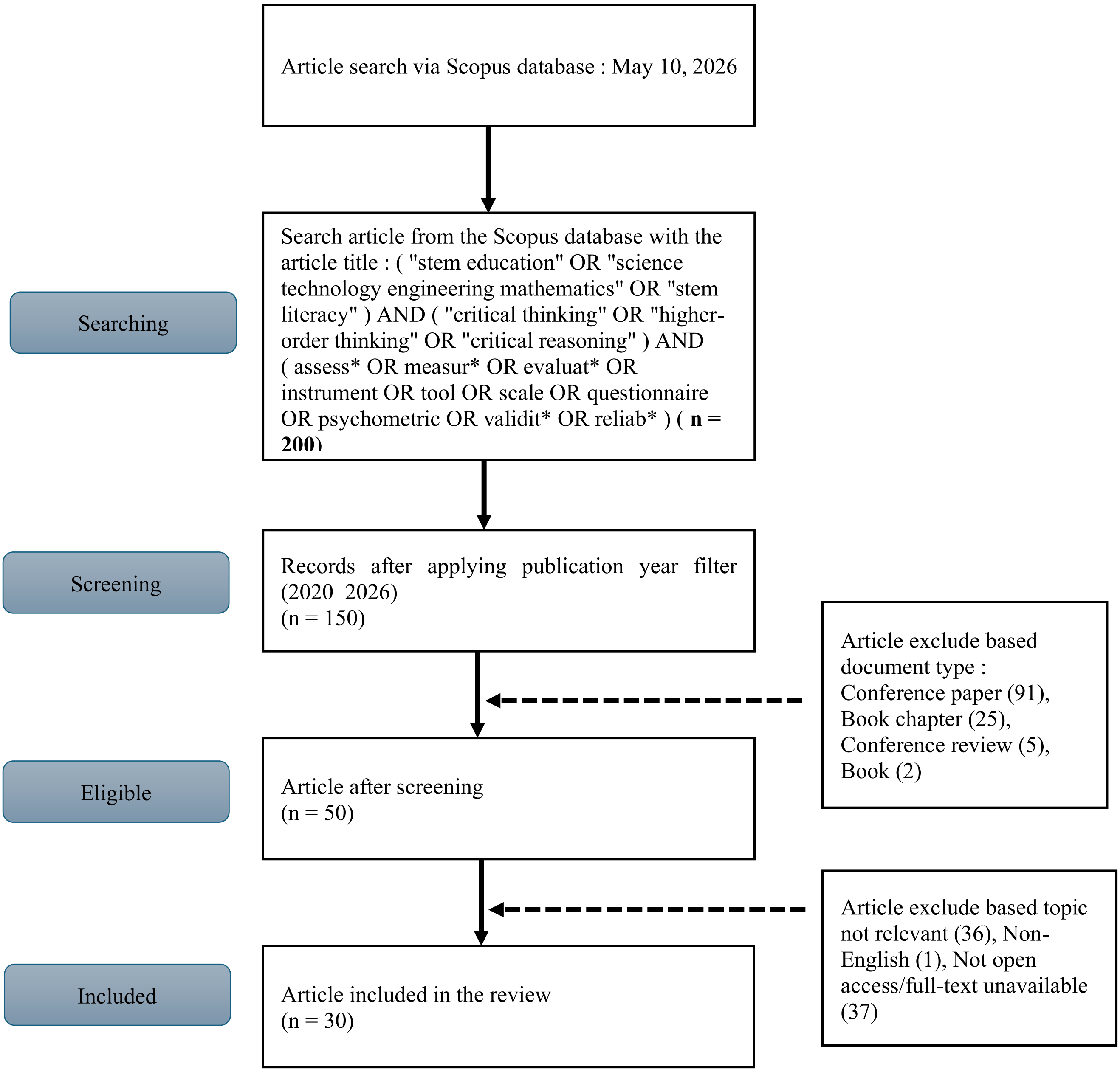

Figure 1. PRISMA Flow Diagram

Results and Discussion

An initial search using Harzing's Publish or Perish yielded 200 articles. After deleting the 20 duplicate articles, there are still 150 articles left for further selection. The selection uses the inclusion criteria and the exclusion narrows the number of articles to 50. The articles are then assessed for quality using quality assessment (QA) standards. The results of the evaluation showed that 30 articles met all the predetermined quality criteria. After selection, the 30 articles were comprehensively reviewed to explore the answers to the research questions that had been processed pre-designed.

Table 1. Conclusion Table from Reviewing the Article

No	Title	Author	Focus Topics	Research Results
1	Developing a Computational Educational Game "Math Squirrel" to Improve the Learning Motivation of 6th Grade Elementary School Students	Hasanah et al., 2025	Game-based learning	Computer-based educational games increase students' motivation to learn math, active engagement, and problem-solving
2	Design Student Worksheet on the Topic of Ratio to Support Students' Computational Thinking Skills	Jayanti et al., 2025	LKS	The worksheet is valid and practical, helping students develop CT through decomposition, abstraction, pattern recognition, and algorithms
3	TPACK-based hybrid learning model design for computational thinking skills achievement in mathematics	Helsa et al., 2023	Hybrid learning	The TPACK-based hybrid model is effective in improving the CT of prospective elementary school teachers and pedagogic skills and content
4	Cross-cultural insights on computational thinking in geometry: Indonesian and Japanese students' perspectives	Prahmana et al., 2024	Geometry, Scratch	Students' CT varies between Indonesia and Japan; use of Scratch supports geometry problem-solving strategies
5	Personal View of Computational thinking ability in mathematics learning of exponents in grade IX	Anonymous, 2025	Exponents	Students with high prior knowledge show complete CT; low weak on algorithms and design
6	Effects of augmented reality application integration with computational thinking in geometry topics	Hanid et al., 2022	Geometry, AR	Integration of AR and CT improves students' visualization, geometry achievement, and problem-solving skills
7	Design mobile computational thinkingintegrated mathematics lessons based on the 5E instructional model	Fang et al., 2025	Mobile CT, 5th model	The CT-based 5E learning model increases student engagement, motivation, critical thinking, and collaboration

8	Exploring the multifaceted roles of mathematics learning in predicting students' computational thinking competency	Lee et al., 2023	CT, Mathematical literacy	Mathematical literacy mediates the relationship between mathematical learning beliefs and CT, demonstrating a strong link between mathematics and CT
9	Computational Thinking in Secondary Mathematics Education with GeoGebra: Insights from an Intervention in Calculus Lessons	Chytas et al., 2024	Calculus, GeoGebra	GeoGebra's integration of CT strengthens students' ability to solve calculus problems and increases their interest in mathematics
10	Mathematical Problem-Solving Ability from the Perspective of the Computational Thinking Approach	Fitdyawati et al., 2025	Problem-solving	High-category students meet 4 out of 5 CT indicators; the intermediate category meets all; Low category only pattern recognition and abstraction
11	Computational Thinking with Scratch: A Tool to Work on Geometry in the Fifth Grade of Primary Education	Molina-Ayuso et al., 2024	Scratch, Geometry	Scratch increases motivation and geometry learning; Students' CT evolves as they interact with software
12	Assessment of Computational Thinking Skills of Upper-Secondary School Students	Rohaeti & Huda, 2025	CT assessment	Rasch-based instruments show the CT ability of Indonesian science and mathematics students well; some gender/age bias items; useful instruments for CT assessment
13	Enhancing Middle School Students' Computational Thinking Competency Through Game-Based Learning	Pan et al., 2024	Game-based CT	GBL improves the CT skills of junior high school students, especially decomposition, algorithms, and pattern recognition
14	Computational Thinking Skills and its Impact on TIMSS Achievement: An Instructional Design Approach	Amirali, 2010	TIMSS, CT	Integration of CT in instructional design improves student performance on TIMSS and improves computational thinking skills
15	Computational Thinking and Mathematics Using Scratch: An Experiment with Sixth-Grade Students	Rodríguez-Martínez et al., 2020	Scratch, Arithmetic	The use of Scratch for arithmetic material improves students' CT and problem-solving skills
16	Junior High School Students' Computational Thinking Ability in Solving Mathematical Problems	Hartawan et al., 2024	CT in problem solving	The ability of CT SMP on decomposition components, patterns, algorithms, abstractions varies; Mathematical factors and ICT literacy affect CT
17	A Learning Trajectory for Developing Computational Thinking in Prospective Mathematics Teachers through Python Programming	Irawan et al., 2025	Python programming, CT	Python-based HLTs improve decomposition, algorithms, pattern recognition, and abstraction of aspiring math teachers
18	Integration of Ethnomathematics in Culturally Responsive STEM Education to Foster Computational Thinking	Lestari et al., 2025	Ethnomath, STEM	Integration of local culture with STEM enhances students' creativity and CT skills

19	Computational Thinking Through Unplugged Computer Science Activities	Mumcu et al., 2023	Unplugged CT	An unplugged approach effectively improves students' CT without a digital device
20	Using Project-Based Problem Solving Activities to Identify CT in Students	Yang et al., 2023	PBL, CT	CT can be evaluated via PBL; Project-based activities help with the development of algorithms, decomposition, abstraction, and patterns
21	Investigating the Impact of Gamification Components on Online Learners' Engagement	Meng et al., 2024	Gamification, CT	Gamification increases engagement and motivation to learn in CT-based math learning
22	Promoting CT via Mathematical Semiotics Analysis	Purwasih et al., 2024	Semiotics, CT	Semiotic analysis helps students understand number patterns and solve mathematical problems with CT
23	Competent Computational Thinking Test (cCTt)	El-Hamamsy et al., 2025	Assessment CT	Valid, reliable, gender-fair cCTt instruments, can be used for longitudinal study of students' CT
24	Research on Independent Study + CT Integration in TPACK Hybrid Learning	Yullys & Helsa, 2023	TPACK, CT	Hybrid learning study with e-learning portal improves the CT of prospective teachers through the pre-class and in-class stages
25	Effects of Scratch-Based Activities on 4th-Grade Students' CT Skills	Piedade & Dorotea, 2022	Scratch, CT	Scratch improves CT and numerical problem-solving skills of grade IV students
26	Computational Thinking and Mathematical Modeling in Quasi-Experimental Study	Musaeus & Musaeus, 2023	Modeling, CT	Computer modeling-based learning improves CT over conventional learning
27	Computational Thinking Skill Level of Senior High School Students Major in Natural Science	Huda & Rohaeti, 2024	CT	Indonesian high school science students have good CT even though they have never received formal CT learning
28	Integration of CT in TIMSS Achievement Context	Amirali, 2010	TIMSS, CT	Integration of CT in instructional design improves student performance on TIMSS and improves computational thinking skills
29	Enhancing Elementary School Students' CT and Programming with Graphic Organizers	Yang & Lin, 2024	CT, Programming	Graphic organizers support the development of CT, abstraction, algorithmics, and student pattern recognition
30	CT Practices in Mathematical Explorations Using Microworlds	Pei et al., 2018	Microworld, Geometry	The use of microworlds for problem solving supports the development of CT explicitly, including decomposition, algorithmics, abstraction, and patterns

Pattern analysis was used, and Bibliometric analysis found that.

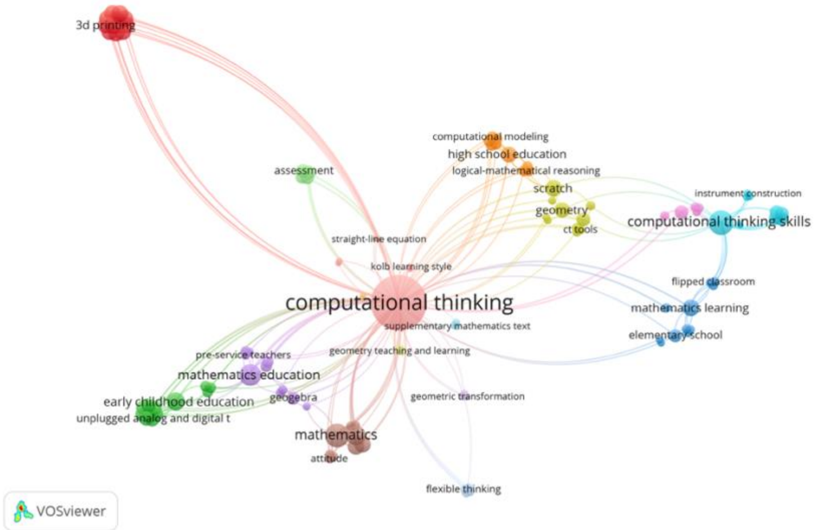

Figure 2. Analysis Bibliometri

The results of a systematic literature review show that integrating *computational thinking* (CT) into mathematics learning plays an important role in learning at the elementary, junior high, and high school levels, particularly in geometry, and that this integration affects students' mathematical problem-solving. CT also directly affects the development of learning technology, such as the use of GeoGebra, 3D Printing, and modelling in computing. CT consists of decomposition, pattern recognition, abstraction, algorithms, and mathematical modelling (Ye et al., 2023; Nordby et al., 2022; Wing, 2006).

A. Analysis of the Impact of Computational Thinking on Mathematics Learning

1. The Impact of CT on Problem Solving and Motivation:

Across the reviewed studies, the contribution of computational thinking (CT) to mathematics learning is not limited to improved performance; rather, CT appears to reorganise how students approach mathematical problems. Decomposition enables learners to reduce complex tasks into manageable elements, pattern recognition supports the identification of structural regularities, abstraction helps filter relevant information, and algorithmic thinking translates reasoning into ordered solution procedures. The stronger performance of students with higher prior knowledge suggests that CT does not operate independently of mathematical foundations; instead, prior conceptual knowledge functions as a condition that enables students to use CT components more completely (Fitdyawati et al., 2025; Rohaeti & Huda, 2025). At the same time, game-based and hybrid TPACK environments appear to strengthen motivation because they provide repeated practice, immediate feedback, and opportunities for active participation. Thus, the cognitive and motivational effects of CT are mutually reinforcing: structured problem-solving supports successful learning experiences, while greater engagement encourages students to persist in computationally demanding mathematical tasks (Hasanah et al., 2025; Helsa et al., 2023).

2. The Impact of CT Integration with Innovative Approaches

The reviewed evidence indicates that the effectiveness of CT depends less on a single instructional model than on whether the model creates opportunities for students to externalise and refine their thinking. Flipped classrooms, 5E, TPACK, mathematical semiotics, ethnomathematics, gamification, and programming-based learning differ in format, yet they share several pedagogical characteristics: active exploration, representation of ideas, iterative problem solving, and reflection. This convergence suggests that CT is best understood as a mode of mathematical reasoning that can be embedded across approaches rather than as a separate technology-based activity. Cultural contexts and real-world problems further strengthen this integration because they give students meaningful situations in which decomposition, pattern recognition, abstraction, and algorithmic reasoning are needed. Consequently, innovative approaches contribute to CT when technology and context serve the mathematical reasoning process, rather than becoming the instructional goal themselves (Lestari et al., 2025; Purwasih et al., 2024).

3. The impact of CT on mathematics learning at the elementary, junior high, and high school levels

At the elementary level, the strongest contribution of CT is its capacity to make abstract mathematical ideas accessible through concrete and interactive experiences. The Math Squirrel study, for example, reported high expert validation and positive student responses, but the more important implication is that game-based tasks allow children to encounter mathematical structure through challenges, visual cues, repetition, and immediate feedback. This interpretation is reinforced by 5E-based mobile CT learning. When elementary students design a simple area-calculation application, they must connect a mathematical formula with input, process, output, testing, and revision.

Such activity changes the status of a formula from something to be memorised into a rule whose logic must be represented and verified. The reported improvements in engagement, motivation, problem solving, critical thinking, communication, and collaboration therefore reflect a broader learning mechanism: CT makes students explain how mathematical knowledge works, not only state the correct answer. For elementary learners, this finding implies that visual, exploratory, and hands-on tasks are especially important because they provide scaffolding for computational reasoning before students are expected to express it formally (Fang et al., 2025).

At the junior high school level, the pattern changes from concrete engagement toward the organisation of increasingly complex reasoning. Hartawan et al. (2024) reported higher performance in abstraction (50.38%) and decomposition (47.25%) than in pattern recognition and generalisation (35.25%) and algorithmic thinking (29.88%). This imbalance is analytically important because it indicates that students may be able to identify relevant information without yet being able to convert that understanding into a complete and sequential solution procedure. In other words, recognising what matters in a problem does not automatically produce algorithmic fluency. Junior high school instruction should therefore emphasise the transition from identifying information to justifying, ordering, and testing solution steps, particularly through tasks that require students to explain why each procedure follows from the previous one (Hartawan et al., 2024).

At the high school level, CT becomes increasingly connected with modelling, representation, and transfer across contexts. Ethnomathematics-based STEM activities show that cultural artefacts can function as mathematically rich problems in which students decompose contextual situations, identify geometric or symmetrical patterns, and formulate systematic solution strategies. The significance of these findings is not simply that students respond positively to local culture, but that contextual familiarity can reduce the initial interpretive burden of a task and allow greater attention to higher-order reasoning. CT at this level therefore serves as a bridge between formal mathematics and complex real-world situations, supporting the movement from procedural calculation toward modelling, justification, and creative problem solving (Lestari et al., 2025).

Taken together, the findings across school levels indicate a developmental progression in the role of CT. In elementary education, CT primarily supports conceptual access through concrete and interactive representation; in junior high school, it strengthens the organisation and sequencing of reasoning; and in high school, it supports modelling, abstraction, and transfer across mathematical and contextual domains. This progression is consistent with the view that mathematical and computational reasoning interact cyclically: mathematics is used to construct computational artefacts, computational processes are used to predict and test mathematical outcomes, and the results generate new mathematical understanding (Ye et al., 2023). Thus, the impact of CT should be evaluated in relation to students' developmental level and the mathematical demands of the task rather than treated as a uniform effect across grades.

4. Impact of CT on students

The synthesis of the reviewed studies suggests that the central student-level effect of CT is a change in the structure of mathematical reasoning. Decomposition, abstraction, pattern recognition, and algorithmic thinking provide students with a sequence for moving from an initially complex problem toward a defensible solution. This is more substantial than simply increasing the number of correct answers because it makes the problem-solving process visible and examinable. Mangiri and Prabawanto (2024) therefore support the interpretation of CT as a metacognitive framework for organising mathematical action: students learn what information to attend to, how to relate it, and how to construct a procedure from it.

However, the benefits of CT are not evenly distributed. Positive emotional responses, persistence, peer collaboration, and repeated independent attempts in digital environments such as Math Squirrel indicate that CT-rich tasks can increase participation and confidence (Hasanah et al., 2025). Yet Hartawan et al. (2024) show that students with lower CT ability continue to experience difficulty in abstraction and algorithmic thinking. These findings reveal an important tension: open and interactive CT activities can increase engagement, but without sufficient scaffolding they may also expose differences in students' readiness. Therefore, motivational design should be accompanied by graduated prompts, worked examples, and opportunities to verbalise reasoning so that students with weaker initial skills are not disadvantaged.

The association between CT and mathematical problem solving is also differentiated by students' proficiency. Fitdyawati et al. (2025) found that students in higher categories demonstrated a more complete set of CT indicators, whereas lower-category students displayed only part of the expected reasoning process. This pattern suggests that CT indicators can function diagnostically, not only as learning outcomes. Teachers can use the absence of a particular component—for example, failure to decompose a problem or to construct an algorithm—to identify where a student's reasoning breaks down. CT assessment can therefore provide more actionable information than a final score because it locates the stage of problem solving that requires intervention.

Mathematical literacy further clarifies why CT varies across students. Lee et al. (2023) found that mathematical literacy predicts CT subscales and mediates the relationship between beliefs about mathematics learning and CT ability. This indicates that CT is embedded in a wider system of knowledge and dispositions: students are more likely to use computational strategies when they can interpret mathematical information and regard mathematics as meaningful and learnable. Consequently, programmes intended to strengthen CT should not isolate computational skills from mathematical literacy or students' beliefs, because weaknesses in either area may constrain the transfer of CT to unfamiliar problems.

5. Impact of CT on math material

Across mathematical topics, CT appears most productive when the content contains structures that can be represented, decomposed, tested, and generalised. Geometry, number operations, number patterns, ratios, measurement, calculus, and modelling all provide such opportunities, but they activate CT in different ways. The prevalence of number operations and geometry in the literature is therefore understandable: these topics make patterns, representations, and procedural relationships comparatively visible (Mangiri & Prabawanto, 2024). The implication is that CT should not be added to mathematics as a generic set of steps; its components should be matched to the epistemic characteristics of each topic.

Geometry illustrates this relationship particularly clearly. Dynamic tools such as Scratch, Jupyter Notebook, GeoGebra, and geometry software allow students to manipulate objects, observe invariant relationships, and test conjectures repeatedly. The analytical value of these tools lies in the feedback loop they create between visual representation and formal reasoning. Students can generate a representation, inspect the result, revise a parameter, and reconsider the underlying mathematical relationship. Consequently, CT in geometry supports not only procedural sequencing but also conjecturing and validation, provided that technological exploration remains explicitly connected to mathematical explanation (Molina-Ayuso et al., 2024; Chytas et al., 2024).

In measurement and area, CT changes formula use from recall to executable reasoning. Creating an area-calculation application requires students to specify variables, translate a formula into a sequence of operations, test inputs, inspect outputs, and correct errors. This process reveals whether students understand the relational meaning of the formula because an incorrect conceptualisation produces an incorrect computational output. Debugging therefore becomes a form of mathematical reflection: students must identify whether an error comes from data entry, procedural sequencing, or conceptual misunderstanding (Fang et al., 2025).

Number and sequence patterns activate CT somewhat differently because the principal challenge is to infer structure and generalise it. Decomposition and abstraction help students isolate relevant terms and relationships, while pattern recognition and algorithmic thinking allow them to formulate a rule that can be applied beyond the given examples. Fitdyawati et al. (2025) consequently show that CT indicators can reveal the quality of reasoning behind a solution, particularly whether students move from observing local regularities to constructing a general procedure. This makes sequence tasks especially useful for assessing the transition from pattern recognition to generalisation.

In calculus, the role of CT is more demanding because students must coordinate symbolic, graphical, and computational representations. GeoGebra can make this coordination visible, but Chytas et al. (2024) caution that successful integration depends on mathematical knowledge and digital competence. This condition prevents a simplistic conclusion that more technology automatically produces better CT. Digital tools become pedagogically productive only when students understand the mathematics sufficiently to interpret computational outputs and when instruction provides time and scaffolding for connecting commands, graphs, and concepts.

6. Impact of CT on teachers or teachers

For teachers, the integration of CT represents a shift from transmitting mathematical procedures to designing environments in which students' reasoning can be elicited, observed, and refined. In a 5E-based mobile CT lesson, for example, the teacher's role spans activating prior knowledge, posing productive challenges, guiding exploration, providing feedback, and evaluating both process and product. This implies that CT changes the object of teaching: teachers must attend not only to whether students obtain a correct result, but also to how they decompose, represent, sequence, test, and revise their reasoning (Fang et al., 2025).

This expanded role requires technological, pedagogical, and content knowledge to operate as an integrated competence. Helsa et al. (2023) show the use of tools such as Microsoft Excel, Unity, R Studio, and Scratch within TPACK-based hybrid learning, but the presence of these tools is not itself evidence of effective CT integration. Their value depends on whether teachers can select a tool because it makes a mathematical relationship, representation, or reasoning process more learnable. Therefore, professional development should prioritise pedagogical decision making around technology rather than training teachers only in software operation.

A further challenge is conceptual. When CT is equated with coding, teachers may overlook mathematically rich unplugged activities that also require decomposition, iterative reasoning, abstraction, and algorithmic planning. The literature therefore suggests that teacher readiness has two dimensions: operational readiness to use digital tools and conceptual readiness to recognise CT in mathematical reasoning itself. Addressing only the first dimension risks producing technology-heavy lessons in which the intended computational reasoning remains implicit (Fitdyawati et al., 2025).

The evidence from GeoGebra-based calculus reinforces this point. Student success depends on didactic design, sufficient familiarisation time, scaffolding, and readiness in mathematical content, indicating that technology acts as a mediating resource rather than an independent cause of learning. Teachers must therefore orchestrate the relationship among mathematical goals, CT practices, and digital affordances. This balance is crucial because excessive attention to the tool can increase cognitive load, whereas well-designed scaffolding can make the tool a means for visualising and testing mathematical ideas (Chytas et al., 2024).

7. The impact of CT on pedagogy

From a pedagogical perspective, the reviewed studies converge on a shift toward active, constructivist, collaborative, and problem-centred learning. Problem-based, game-based, cooperative, project-based, mobile, and unplugged approaches are diverse, yet each can support CT when students are required to make their reasoning explicit and revise it through feedback. Simulations, programming platforms, visualisation tools, and interactive games are therefore best treated as environments for inquiry rather than as ends in themselves (Mangiri & Prabawanto, 2024).

The 5E model provides a useful illustration of how CT can be embedded systematically. Engagement activates prior knowledge; exploration creates space for decomposition and testing; explanation requires students to articulate patterns and abstractions; elaboration extends those ideas to more complex situations; and evaluation examines both outcomes and reasoning processes. The analytical significance of this alignment is that CT is distributed across the learning cycle instead of being confined to a final coding activity. Such integration makes assessment more process-oriented because students' decisions, revisions, and explanations become visible evidence of learning (Fang et al., 2025).

The claim that CT supports differentiation and inclusion must nevertheless be interpreted cautiously. Adaptive and interactive games such as Math Squirrel can provide self-paced practice and immediate feedback, which may reduce anxiety and support students with different learning speeds (Hasanah et al., 2025). However, inclusivity depends on access to devices, the readability of representations, and the availability of scaffolding. CT-based pedagogy is therefore potentially inclusive, but only when differences in digital access and prior knowledge are explicitly accommodated in instructional design.

Hybrid TPACK learning extends this pedagogical logic beyond the physical classroom by combining independent preparation, face-to-face discussion, digital activity, quizzes, feedback, and evaluation. Its main contribution is not hybridity itself but the possibility of distributing different CT processes across time and learning spaces. Students can first encounter information independently, use classroom interaction to compare strategies, and then employ digital tools to test or refine solutions. This suggests that sustainable CT pedagogy requires coherence among pre-class, in-class, and post-class activities rather than the simple addition of an LMS or digital platform (Helsa et al., 2023).

8. The impact of CT on the school environment

At the school level, CT can influence classroom culture by encouraging collaboration, experimentation, and technology-supported problem solving. Evidence from local culture-based STEM learning and Math Squirrel suggests that students become more active when tasks invite exploration and shared problem solving, but these outcomes also depend on organisational support (Lestari et al., 2025; Hasanah et al., 2025). The school-level effect is therefore conditional: innovation in classroom practice requires access to devices, technical support, teacher guidance, and sufficient opportunities for professional learning. CT should consequently be understood as both a pedagogical innovation and an organisational capacity-building agenda.

International variation in CT integration further indicates that there is no single institutional model. CT may be positioned within informatics, programming, mathematics, or interdisciplinary learning, but each arrangement requires curricular coherence and clear learning outcomes (Suarsana et al., 2024). For schools, the strategic issue is therefore not where CT is administratively placed, but whether students encounter progressively more complex opportunities to apply computational reasoning across subjects. A supportive school environment should connect curriculum, teacher competence, infrastructure, and assessment rather than treating CT as an isolated programme.

Equity is a final institutional consideration. Integrating CT into compulsory mathematics can broaden participation because all students encounter mathematics, including learners who might not voluntarily enrol in computing-related subjects (Chytas et al., 2024). However, compulsory access does not automatically guarantee equitable learning. Differences in device availability, digital familiarity, teacher support, and prior mathematical knowledge can still shape outcomes. Schools therefore need to combine broad curricular access with targeted support so that CT integration expands participation without reproducing existing digital and academic inequalities. Overall, the reviewed literature indicates that CT has its strongest impact when student readiness, mathematical content, pedagogy, teacher competence, and school infrastructure operate as an interconnected system rather than as independent factors.

B. Analysis of Success Factors for the Application of CT in Mathematics Learning

The first factor is teacher readiness and conceptual understanding of CT. Teachers need to understand that CT is not just coding, but a way of thinking that includes decomposition, abstraction, pattern recognition, and algorithmic thinking. If teachers still interpret CT as just programming, its application will be limited to using applications, not to developing students' thinking processes. Therefore, teacher training, strengthening TPACK, and providing examples of CT-based math assignments are the main requirements for success (Helsa et al., 2023; Fitdyawati et al., 2025).

The second factor is a systematic and student-level learning design. CT is successfully applied when learning is designed in stages, starting from initial knowledge activation, problem exploration, solution development, testing, and reflection. The 5E model provides a concrete example of how CT can be designed in a clear pedagogical structure, so that students not only use technology, but also understand mathematical concepts through computational processes (Fang et al., 2025).

The third factor is the selection of media that is appropriate and relevant to the learning objectives. Media such as Scratch, GMedia such as Scratch, GeoGebra, App Inventor, AR, educational games, Excel, RStudio, and unplugged activities can be successful if chosen based on the material, students' ages, and the availability of facilities.asised the importance of choosing computational tools by considering students' knowledge levels, teaching needs, adaptation time, and tool effectiveness (Fang et al., 2025).

The fourth factor is the suitability of the mathematical material with the characteristics of CT. Materials containing patterns, procedures, visual representations, data, and modelling are easier to integrate with CT. Geometry, number operations, number patterns, area, ratios, and calculus are powerful materials because they allow students to visualise, decompose, generalise, and compile algorithms. However, the SLR article also shows that there is still an opportunity to develop CT in other materials that have not been widely researched (Mangiri & Prabawanto, 2024; Ye et al., 2023).

The fifth factor is scaffolding and support for students' different abilities. Students with low CT ability need help in understanding important information, recognising problems, and devising solutions. Data from Hartawan et al. (2024) show that algorithmic thinking is the lowest component, so teachers need to provide examples of completion steps, guiding questions, step-by-step exercises, and feedback so that students not only find answers but also understand their thinking process (Hartawan et al., 2024).

The sixth factor is a valid CT assessment, not just an assessment of the final answer. Rohaeti and Huda (2025) show that CT assessments can be developed using the Rasch model and help teachers and researchers more accurately assess CT abilities in high school students. However, assessments also need to take into account possible gender biases, students' ages, and the background of their experiences. With proper assessment, teachers can identify which components of CT are already strong and which need improvement (Rohaeti & Huda, 2025).Support for school infrastructure and culture. Implementing CT requires a school environment that supports technology, collaboration, experimentation, and innovation. Math Squirrel articles show that digital media developed with Unity require an application format, user guidance, teacher orientation, and dissemination for use in a school context. Therefore, the success of CT depends not only on teachers and students but also on schools' readiness to provide tools, networks, time, and policy support (Hasanah et al., 2025).

The eighth factor is the contextualization of learning. CT is more successful when it is associated with a student's real life, local culture, authentic issues, or meaningful projects. The integration of ethnomathematics in STEM shows that cultural elements such as batik, bamboo weaving, and traditional units can make CT learning more relevant and humane. Learning like this helps students see that mathematics and CT are not distant from life but are present in culture, the environment, and daily activities (Lestari et al., 2025).

The ninth factor is the strengthening of mathematical literacy as the foundation of CT. Lee et al. (2023) show that math literacy mediates the relationship between math learning beliefs and CT competencies. This means that CT success cannot be separated from the student's ability to understand mathematical concepts, use representations, and interpret problems meaningfully (Lee et al., 2023).

The tenth factor is the critical use of technology, not passively. Yunianto et al. (2025) show that ChatGPT's integration with GeoGebra can open up opportunities for CT development, but students still need to be trained to analyse, improve, and debug AI outputs, rather than receiving answers directly. Thus, the success of CT in the AI era depends on teachers' ability to build a learning environment that encourages students' active engagement and critical reflection (Yunianto et al., 2025).f CT in mathematics learning is determined by a combination of teacher understanding, pedagogical design, appropriate media, appropriate materials, student support, valid assessments, school infrastructure, mathematical literacy, and meaningful learning contexts. CT will have the greatest impact if it is not treated as a mere technological add-on, but as a thinking approach integrated with mathematics learning goals, student needs, and school realities (Ye et al., 2023; Mangiri & Prabawanto, 2024; Hasanah et al., 2025).

C. Challenges and Implications:

1. Challenges include teacher readiness, design quality, access to technology, and valid assessment instruments (Lee et al., 2023; Nordby et al., 2022).
2. Implementation. Students' prior knowledge influences the success of CT development; Less experienced students tend to struggle with algorithmic and solution design (Zhang et al., 2024; Fitdyawati et al., 2025).

Conclusion and Suggestion

This systematic literature review concludes that computational thinking (CT) has a significant and positive impact on mathematics learning. CT is not just programming or the use of digital technology, but rather a structured way of thinking that helps students break down complex problems, recognise patterns, abstract important information, formulate algorithms, test solutions, and revise errors. These components are closely related to mathematical thinking because mathematics requires logical reasoning, problem decomposition, modelling, representation, and systematic problem-solving (Wing, 2006; Weintrop et al., 2016; Ye et al., 2023; Nordby et al., 2022).

The study's results show that CT supports mathematics learning at the elementary, junior high, and high school levels. CT helps students understand mathematical concepts in a concrete, visual, interactive, and fun way, especially through educational games, mobile learning, and visual programming activities (Hasanah et al., 2025; Fang et al., 2025). At the junior high school level, CT strengthens students' ability to solve mathematical problems more systematically, although students still need support in algorithmic thinking and abstraction (Hartawan et al., 2024; Fitdyawati et al., 2025). At the high school level, CT contributes to higher-level thinking skills, particularly in geometry, calculus, STEM-based learning, mathematical modelling, and contextual problem-solving (Chytas et al., 2024; Lestari et al., 2025).

The impact of CT on students includes improved cognitive and affective abilities. Cognitively, CT improves critical thinking, logical reasoning, problem-solving, abstraction, pattern recognition, and algorithmic thinking. Affectively, CT can increase student motivation, confidence, perseverance, and engagement, especially when integrated with game-based learning, gamification, project-based learning, ethnomathematics, GeoGebra, Scratch, and mobile learning (Hasanah et al., 2025; Molina-Ayuso et al., 2024; Meng et al., 2024; Lestari et al., 2025). However, the impact of CT is not always evenly distributed, as early knowledge, math literacy, digital literacy, and experiential learning affect students' ability to apply CT effectively (Lee et al., 2023; Hartawan et al., 2024; Zhang et al., 2024).

From a teacher and pedagogical perspective, CT drives a shift from teacher-centred learning to more active, exploratory, collaborative, and problem-based learning. Teachers not only deliver material but also design meaningful assignments, provide scaffolding, facilitate reflection, choose appropriate media, and assess students' thinking processes. Therefore, teacher readiness, TPACK competence, pedagogical design, and conceptual understanding of CT are important factors in integrating CT into mathematics learning (Helsa et al., 2023; Fang et al., 2025; Fitdyawati et al., 2025).lementation of CT is determined by a combination of factors, namely teacher understanding, learning design, appropriate media, appropriate materials, student support, valid assessments, school infrastructure, mathematical literacy, and meaningful learning contexts. CT is more effective when it is an integral part of mathematics learning, rather than just a tool. Activities that combine math with culture, STEM, real-world problems, digital tools, or unplugged activities can make CT more inclusive, relevant, and accessible to students (Mangiri & Prabawanto, 2024; Ye et al., 2023; Lestari et al., 2025; Mumcu et al., 2023).

However, challenges in implementing CT include limited teacher understanding, misconceptions that CT is synonymous with coding, uneven access to technology, limited implementation time, variations in students' initial knowledge, and the absence of valid and practical CT assessment instruments. Therefore, future mathematics learning needs to systematically integrate CT through curriculum design, teacher training, CT-based material development, and assessment instruments that assess not only the final answer but also the student's thought process (Rohaeti & Huda, 2025; El-Hamamsy et al., 2025; Chytas et al., 2024).

Overall, computational thinking is an important competency to strengthen mathematics learning in the 21st century. CT helps students move beyond procedural calculations toward deeper mathematical reasoning, reflective problem-solving, creativity, and adaptability. Therefore, CT should be placed as an integral part of mathematics education, not just an additional skill. Follow-up research is recommended to conduct empirical, longitudinal, and classroom-based studies to examine the long-term impact of CT integration on mathematical achievement, problem-solving ability, digital literacy, and readiness to deal with complex real-world problems (Wing, 2006; Weintrop et al., 2016; Ye et al., 2023; Nordby et al., 2022).

References

Cahdriyana, & Richardo, R. (2020). *Mathematics problem-solving and computational thinking skills in Indonesia*. Jakarta: University of Indonesia Press.

Chytas, C., van Borkulo, S. P., Drijvers, P., Barendsen, E., & Tolboom, J. L. J. (2024). Computational Thinking in Secondary Mathematics Education with GeoGebra: Insights from an Intervention in Calculus Lessons. *Digital Experiences in Mathematics Education, 10*, 228–259.

Fitdyawati, S. D., In'am, A., & Zukhrufurrohmah, Z. (2025). Mathematical Problem-Solving Ability from the Perspective of the Computational Thinking Approach. *Creano, 16*(1), 210–227.

Hartawan, I. G. N. Y. (2024). Junior high school students' computational thinking ability in solving mathematical problems. *Journal of Pedagogy and Learning, 7*(1), 124–133.

Hasanah, H., Abadi, M. K., Yanuar, M., Darmawan, R., & Gandhi, H. K. (2025). Developing a Computational Educational Game "Math Squirrel" to Improve the Learning Motivation of 6th Grade Elementary School Students. *Creano, 16*(1), 136–153.

Helsa, Y., Turmudi, & Juandi, D. (2023). TPACK-based hybrid learning model design for computational thinking skills achievement in mathematics. *Journal on Mathematics Education, 14*(2), 225–252.

Hendriyani, D. (2020). Algorithmic thinking in mathematical problem-solving. *JP2: Journal of Pedagogy & Learning, 7*(1), 124–133.

Jayanti, R. D., Mulyono, B., Hapizah, H., & Cahyawaty, D. (2025). Design Student Worksheet on the Topic of Ratio to Support Students' Computational Thinking Skills. *Creano, 16*(1), 154–170.

Lehmann, T. H. (2024). Mathematical modelling as a vehicle for developing algorithmic thinking. *Educational Studies in Mathematics, 115*(2), 151–176.

Lestari, D., Dafik, A., & Isyati, T. (2025). Integration of Ethnomathematics in Culturally Responsive STEM Education to Foster Computational Thinking. *Journal of Educational Research, 10*(2), 341–362.

Looi, C.-K., Chan, S.-W., Wu, L., Huang, W., Kim, M. S., & Sun, D. (2024). Exploring computational thinking in the context of mathematics learning in secondary schools: Dispositions, engagement and learning performance. *International Journal of Science and Mathematics Education, 22*(5), 993–1011.

Mangiri, G. R. P., & Prabawanto, S. (2024). Exploring computational thinking in learning mathematics: A systematic literature review from 2017–2024. *Journal of Mathematics and Natural Sciences Education, 25*(1), 34–52.

Molina-Ayuso, A., Adamuz-Povedano, N., Bracho-López, R., & Torralbo-Rodríguez, M. (2024). Computational Thinking with Scratch: A Tool to Work on Geometry in the Fifth Grade of Primary Education. *Education Tech Research Dev, 73*, 2809–2838.

Ng, O.-L., Leung, A., & Ye, H. (2023). Exploring computational thinking as a boundary object between mathematics and computer programming for STEM teaching and learning. *ZDM*, 55(7), 1315–1329.

Nordby, S. K., Bjerke, A. H., & Mifsud, L. (2022). Computational Thinking in the Primary Mathematics Classroom: A Systematic Review. *Digital Experiences in Mathematics Education, 8*, 27–49.

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.

Roth, E., & Huda, M. K. (2025). Assessing the computational thinking skills of upper-secondary school students. *Journal of Educational Research, 13*(4), 200–225.

Urhan, S., & Kocadere, S. A. (2024). Problem-solving through *pair-programming*: The mediational role of ChatGPT. *2024 5th International Conference in Electronic Engineering, Information Technology & Education (EEITE),* 1–10.

Wardat, Y., Tashtoush, M. A., AlAli, R., & Jarrah, A. M. (2023). ChatGPT: A revolutionary tool for teaching and learning mathematics. *Eurasia Journal of Mathematics, Science and Technology Education, 19*(7), em2286.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jonah, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology, 25*(1), 127–147.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35.

Ye, H., Liang, B., Ng, O.-L., & Chai, C. S. (2023). Integration of computational thinking in K-12 mathematics education: a systematic review. *International Journal of STEM Education, 10*, 3.

Zhang, X., Li, H., & Wang, Y. (2024). Prior knowledge and computational thinking in mathematics. *Education and Information Technologies, 29*, 345–362.