2025, 6(3), 197–206

Journal of Didactic Mathematics
https://doi.org/10.34007/jdm.v6i3.2937

Computational thinking obstacles in students' responses to AKM level 5 problems

Ginda Maruli Andi Siregar*, [a], Hidayat [b], Sukmawarti [b], Aya Shofia Maulida [c]
[a]Universitas Samudra, Langsa, Aceh, Indonesia, 24416
[b]Universitas Muslim Nusantara Al-Washliyah, Medan, North Sumatra, Indonesia, 20147
[c]Universitas Pendidikan Indonesia, Bandung, West Java, Indonesia, 40154

Abstract.
The objective of this research is to elucidate the impediments encountered by students when attempting to solve Level 5 Minimum Competency Assessment (AKM) problems within the realm of Computational Thinking (CT). Employing a descriptive qualitative approach with a case study design, the research involved eleventh-grade students from a senior high school. Data were gathered through Level 5 AKM tests, in-depth interviews, and the analysis of students' written responses. These data were subsequently analyzed using open coding, selective coding, and axial coding. The findings reveal that students encounter CT obstacles in several critical domains. Specifically, in the decomposition indicator, students demonstrated difficulties in breaking down graphical information, selectively extracting data without comprehending the interrelationships among values. In pattern recognition, students failed to discern upward-downward trends in harvest data or proportional relationships in probability tasks, thereby hindering their ability to draw comprehensive conclusions. Abstraction challenges emerged when students were unable to discern pertinent information, such as conflating actual frequencies with theoretical probabilities. In algorithmic thinking, students were unable to construct systematic steps in calculations or engage in logical reasoning. Furthermore, logical reasoning and evaluation were deficient, as evidenced by their inability to assess the plausibility of results or validate their answers.

Keywords:
Computational thinking; obstacles; minimum competency; case study; descriptive qualitative

How to cite:
Siregar, G. M. A., Hidayat, H., Sukmawarti, S., & Maulida, A. S. (2025). Computational thinking obstacles in students' responses to AKM level 5 problems. *Journal of Didactic Mathematics*, *6*(3), 197–206. https://doi.org/10.34007/jdm.v6i3.2937

INTRODUCTION

The Minimum Competency Assessment (AKM) serves as a pivotal component of the National Assessment, meticulously crafted by the Ministry of Education, Culture, Research, and Technology (Kemendikbudristek, 2021). This assessment system aims to establish a more precise evaluation framework that comprehensively measures students' fundamental competencies. The development of AKM was necessitated by the imperative to transition from content-oriented evaluation to competency-based measurement. This shift underscores the importance of assessing students' proficiency in comprehending texts, employing logical reasoning, analyzing data, and solving practical challenges (Pusmenjar, 2020). The AKM's development draws upon international assessments such as the Program for International Student Assessment (PISA) and the Trends in International Mathematics and Science Study (TIMSS), expert consultations, instrument validation, and the construction of indicators representing literacy and numeracy. These indicators serve as the foundation for fostering lifelong learning (OECD, 2019).

* Corresponding author.
E-mail address: ginda.mas@unsam.ac.id

Received: 24 November 2025; Revised: 12 December 2025; Accepted: 31 December 2025
2721-5601/ © 2025 The Author(s). Published by Mahesa Research Center

The primary objective of the Assessment of Knowledge and Learning (AKM) is to provide diagnostic information that schools can utilize to enhance instructional practices. AKM is not intended as a graduation requirement or selection tool but rather as a system-level indicator of student learning at the school, regional, and national levels (Pusmenjar, 2020). Consequently, AKM items are designed to assess students' ability to apply concepts rather than merely recall formulas (Wijaya et al., 2015). This aligns with the demands of the 21st century, where reasoning, data literacy, and problem-solving are fundamental competencies (Partnership for 21st Century Learning, 2019).

Following the development of the framework, the government conducted national socialization and field trials. AKM was officially implemented in 2021 as part of the National Assessment, replacing the National Examination (Kemendikbudristek, 2021). Implementation encompassed stakeholder training, CBT infrastructure preparation, and the development of teacher support materials (Pusmenjar, 2020). However, many schools initially encountered challenges such as limited digital facilities, inadequate teacher comprehension, and students' unfamiliarity with context-rich assessment items (Ahmad, 2022). These systemic changes introduced a novel assessment experience that many students found challenging. Students were often unaccustomed to handling lengthy texts, intricate data representations, and realistic scenarios that demanded higher-order reasoning (Rochmaeni & Wardana, 2023). Teachers also required adjustments to their instructional strategies to prioritize reasoning and mathematical interpretation (Bioto et al., 2022). This resulted in a disparity between AKM expectations and students' actual competencies.

In the AKM Numeracy Level 5 curriculum, elements of Computational Thinking (CT) are explicitly integrated. CT encompasses the capacity to break down problems, discern patterns, identify pertinent information (abstraction), establish systematic procedures (algorithmic thinking), and derive conclusions based on logical relationships (Shute et al., 2017; Wing, 2006). AKM tasks necessitate students to interpret contexts, analyze graphs, synthesize multiple data points, and select efficient solution strategies (Kemdikbud, 2020). Consequently, AKM assessments not only evaluate numeracy skills but also computational thinking capacities (Angeli & Giannakos, 2020).

Field observations indicate that students frequently encounter challenges when applying computational thinking (CT) to solve Advanced Knowledge and Mastery Level 5 (AKM Level 5) tasks. Previous research suggests that students struggle to decompose problems, extract essential information, and identify underlying patterns, hindering their ability to construct systematic strategies (Ashiddiqi et al., 2024; Pusat Asesmen Pendidikan, 2024). These difficulties are further exacerbated by limited teacher knowledge of AKM structure and the absence of explicit instruction that develops computational thinking frameworks (Rochmaeni & Wardana, 2023). Additionally, students demonstrate weaknesses in interpreting complex information, comprehending data representations, and validating solutions—indicating deficits in algorithmic thinking and debugging (Ashiddiqi et al., 2024). These findings underscore the necessity of analyzing student difficulties in AKM through a computational thinking framework rather than solely content-focused analysis (Barr & Stephenson, 2011). CT indicators such as decomposition, abstraction, pattern recognition, algorithmic thinking, and debugging constitute fundamental cognitive foundations for interpreting information and selecting appropriate solution approaches (Grover & Pea, 2013). Consequently, an in-depth examination of students' computational thinking obstacles is imperative to design targeted instructional interventions and enhance literacy and numeracy as the primary objectives of the national assessment.

METHOD

This study employed a qualitative descriptive case study design to analyze students' computational thinking (CT) obstacles when solving Asesmen Kompetensi Minimum (AKM) problems. A total of 30 students from SMA Negeri 1 Kota Langsa were selected through purposive sampling to complete two AKM narrative tasks consisting of four questions released by PUSMENDIK in the domain of numerical literacy. Each question required multiple responses across 11 analytical statements. The research instruments included AKM questions, student worksheets, a semi-structured interview guide, and a rubric of Computational Thinking indicators covering Decomposition, Pattern Recognition, Abstraction, Algorithmic Thinking, Data Representation, Debugging, and Verification (Angeli & Giannakos, 2020; Wing, 2006). As depicted in Table 1.

Data collection was conducted in three distinct stages: (1) administering the AKM test to all students under controlled conditions; (2) conducting semi-structured interviews to elucidate students' thought processes, challenges, and rationales for their answer selections; and (3) documenting interviews, field notes, and student worksheets. A select group of participants also engaged in a straightforward think-aloud session to observe their real-time reasoning. All written and verbal data were triangulated to enhance the validity of the findings, with the researcher assuming the primary role as the instrument (Creswell, 2014).

Table 1. CT indicator from AKM problem

Core Question	Implied Sub-Questions	CT Indicator
Evaluating true–false statements from a harvest graph	In which month is the smallest production increase?	Pattern Recognition – Identifying patterns of increase/decrease in the graph.
	Does the order of production increases (largest → smallest) match the statement?	Decomposition – Breaking the graph into monthly segments for analysis.
	Is the production drop in April greater than in June?	Algorithmic Thinking – Computing month-to-month changes in sequence.
		Abstraction – Selecting only the necessary months for comparison.
Selecting correct statements from a rice-harvest graph	Is the increase in production from July to August greater than the previous month?	Pattern Recognition – Reading trends from visual points on the graph.
	Is the harvest from March–August less than ½ of total 2019 harvest?	Data Representation – Converting the graph into estimated values and comparing totals.
	Is the decrease in April equal to the decrease in June?	Algorithmic Thinking – Calculating month-to-month differences.
	Is August's harvest four times December's?	Decomposition – Taking two distinct data points to compare.
		Verification – Checking the accuracy of comparisons with the graph.
Evaluating a commentator's opinion from a probability table	Who has the highest chance of becoming champion?	Pattern Recognition – Identifying the largest probability value in the column.
	Can Team B meet Team A in the final?	Algorithmic Thinking – Following the path: semifinal → final.
	Do Teams G and H have equal chances of reaching the final?	Abstraction – Using only the probability-to-final column.
		Data Representation – Relating table values to the competition bracket.
Determining the team most likely to meet H in the final	If H is guaranteed to reach the final, which team from Matches 1 & 2 has the greatest chance to face them?	Decomposition – Separating early matches → semifinal → final.
		Algorithmic Thinking – Calculating the step-by-step chance toward the final.
		Pattern Recognition – Detecting the highest probability to the final.
		Abstraction – Ignoring irrelevant information (e.g., probability of becoming champion).

Data analysis in this study was conducted through three primary stages: open coding, axial coding, and selective coding. These stages adhered to qualitative analysis procedures grounded in the principles of grounded theory (Corbin & Strauss, 2008; Creswell, 2014).

In the open coding stage, all data, including students' written responses, interview transcripts, and field notes, were repeatedly examined to identify units of meaning such as misreading information, inability to distinguish relevant data, disorganized solution steps, or the absence of self-checking processes. Each finding was assigned an initial code without predetermined categories (Corbin & Strauss, 2008).

In the axial coding stage, these codes were organized into more structured categories based on Computational Thinking indicators—Decomposition, Pattern Recognition, Abstraction, Algorithmic Thinking, Data Representation, Debugging, and Verification—allowing patterns to emerge that connect students' errors with weaknesses in specific CT processes.

Finally, in the selective coding stage, core categories were selected and integrated to formulate overarching themes that represent students' computational thinking obstacles holistically.

RESULTS AND DISCUSSION

The analysis delved into the students' Computational Thinking (CT) capabilities by examining their responses to AKM mathematical literacy problems. The review was conducted by analyzing how each CT indicator manifested in the students' problem-solving processes, encompassing problem decomposition, pattern recognition, abstraction, and algorithmic thinking in devising solution steps. The results of the categorization based on the words employed in students' written answers and interviews are presented in Table 2 as follows.

Table 2. Categories of Students' Responses Based on Open and Selective Coding

No.	Category	Freq.	Open Coding (Student Expressions)	Selective Coding
1	Confusion / Lack of Understanding	32	confused, don't know, don't understand, unclear, vague, not visible, dizzy, difficult, hard, unfocused, failed, blank	Inability to understand the problem (CT: weak abstraction)
2	Misinterpretation of Data / Graph	21	misread, misinterpret, wrong value, wrong graph, wrong pattern, wrong probability interpretation, reversed	Incorrect data interpretation (CT: low data representation)
3	Lack of Stepwise Process / Non-Algorithmic	26	random, guess, trial-and-error, assume, no steps, unordered, no calculation, no checking, no analysis	Absence of systematic procedures (CT: weak algorithmic thinking)
4	Errors in Reading Visual Data	18	did not read graph, did not read table, no connection, no comparison, missing data, cannot find	Failure to process visual representations (CT: weak data representation)
5	Inaccuracy / Lack of Verification	12	careless, doubtful, not confident, inaccurate, unsure	Lack of validation (CT: weak evaluation & verification)
6	Errors in Probability Concepts	17	wrong probability, big chance? small chance? don't understand probability, miscalculation	Weak probability understanding (CT: weak abstraction & logical reasoning)
7	Uncertainty / Inconsistent Reasoning	8	maybe, seems like, not sure, illogical	Instability in reasoning (CT: weak logical reasoning)
8	Indication of Pattern Recognition	13	increase, decrease, upward pattern, downward pattern, change, relationship, compare, order	Pattern identification (CT: pattern recognition)
9	Indication of Data Understanding	9	highest, lowest, appropriate, correct, accurate, consistent, big value, small value	Understanding and comparing values (CT: strong abstraction)

No.	Category	Freq.	Open Coding (Student Expressions)	Selective Coding
10	Analytical Actions / Good Understanding	8	analyze, careful, identify, conclude, represent, re-check	Analytical thinking emergence (CT: algorithmic thinking)
11	Visual Data Representation (Graph/Table)	7	graph, diagram, table, matching data, visual	Ability to interpret visual data (CT: data representation)
12	Inability to Conclude	6	no conclusion, inconsistent, mismatch	Failure to create final conclusion (CT: weak abstraction)
13	Information Overload / Key Feature Failure	5	too much data, complex data, cannot see changes	Difficulty decomposing information (CT: weak decomposition)

Utilizing the categorized open coding and selective coding results, axial coding was employed to establish a connection between the various categories of obstacles and the Computational Thinking (CT) abilities demonstrated in students' responses. The axial coding elucidates how students' statements reflect either deficiencies or strengths in CT components. The connections between selective coding for each question are delineated in the axial coding presented in Table 3 below.

Table 3. Axial Coding Based on Students' Responses to AKM Items

Item	Selective Coding	Axial Coding
1. Harvest Graph – Reading Data	Difficulty interpreting the graph	Students extracted information only from isolated segments of the graph without understanding the relationships among values. They failed to connect axes, trends, and comparisons. Errors stemmed from weak *decomposition* and poor *data representation*. Their understanding of the graph was partial, leading to inaccurate interpretations and showing that foundational CT skills were not yet formed.
2. Harvest Graph – Drawing Conclusions	Inability to derive conclusions from the graph	Students tended to draw conclusions based on a single data point, ignoring overall trends of increase or decrease. Weak *pattern recognition* prevented them from constructing logical inferences from the available data. Their failure to integrate information across the entire graph resulted in incomplete conclusions, indicating underdeveloped CT skills.
3. Basic Probability – Determining Probabilities	Insufficient understanding of probability concepts	Students struggled to construct sample spaces, confused actual frequencies with theoretical probabilities, and used incorrect numerical reasoning. Weaknesses in *abstraction* and *modeling* prevented them from translating random situations into mathematical form. This reflects the absence of a systematic CT structure in their reasoning.
4. Probability – Comparing Probabilities	Inaccurate probability reasoning	Students were unable to compare probabilities correctly due to misunderstandings of event–likelihood relationships and frequent errors in applying numerator–denominator logic. Weak *algorithmic thinking* and poor *evaluation* skills resulted in failure to assess the plausibility of their answers. These issues highlight insufficient CT competence in probabilistic reasoning.

Through this categorization, it is evident that the majority of students' difficulties are centered on their inability to comprehend problems, errors in interpreting visual data, and the absence of systematic steps in problem-solving. Conversely, some categories indicate the presence of pattern recognition skills and appropriate data processing abilities. The weaknesses observed in solving graph and probability problems do not appear in isolation but are interrelated, forming a consistent pattern of cognitive obstacles. The findings reveal that students still struggle with decomposition, abstraction, pattern recognition, data representation, algorithmic thinking, and verification when processing information and drawing conclusions. The detailed weaknesses faced by students are described in each category as follows:

Decomposition (Problem Decomposition)

The findings reveal that a significant portion of students encounter challenges in decomposing information into simpler components when confronted with graph and probability-based problems. In graph-related tasks, students demonstrate difficulties in identifying crucial elements such as annual crop values, trends of increase or decrease, and comparisons between data points. Conversely, in probability problems, students fail to systematically break down sample spaces, potential events, or calculation steps, resulting in unorganized problem-solving. Open coding revealed words such as “confused,” “don’t know,” “difficult,” “misread graph,” and “no steps,” indicating students’ unpreparedness for step-by-step problem-solving. Selective coding corroborated that these difficulties reflect an inability to deconstruct information structures, while axial coding demonstrated that most students process information globally, overlooking significant details. This lack of decomposition emerges as a substantial impediment to subsequent cognitive processing stages, including pattern recognition, abstraction, and algorithmic thinking.

The deficiency in decomposition hinders students from developing a systematic problem-solving strategy, often leading to inaccurate responses. These findings align with the notion that decomposition is a fundamental component of cognitive processing (Shute et al., 2017; Wing, 2006). Students frequently perceive graphs as mere visual representations rather than data analysis tools (Sari et al., 2023). Conversely, suggests that decomposition can be enhanced when students receive scaffolding or step-by-step instructions, emphasizing the significant influence of the learning context on this ability (Rahmawati et al., 2024). Consequently, the development of decomposition skills necessitates instructional strategies that facilitate stepwise analysis and explicit identification of essential elements.

Pattern Recognition

Students’ ability to discern patterns remains severely constrained in both graph and probability-based tasks. In graph-related tasks, a significant portion of students fail to identify upward and downward trends over time and overlook relationships between data points. In probability-based tasks, they encounter difficulties in detecting patterns that connect sample spaces, events, and probabilities, including proportionality patterns that underpin probability calculations. Open coding revealed terms such as “cannot see pattern,” “no comparison,” “wrong trend,” and “cannot find connection,” while words like “up,” “down,” or “change” were rarely encountered. Selective coding confirmed students’ inability to recognize patterns visually or numerically, whereas axial coding demonstrated that this inability leads to erroneous inferences and illogical conclusions. A limited pattern recognition ability directly impacts the construction of mental representations and the generalization of information derived from data.

The restricted pattern recognition capacity causes students to struggle with connecting data, observing changes, and forming generalizations from graphs and probability problems. Consequently, their responses tend to be random, inconsistent, and devoid of data-based reasoning. The absence of this ability also affects subsequent cognitive tasks, such as algorithmic thinking and evaluation, as students lack a logical foundation to guide the subsequent steps. Without pattern recognition skills, students are susceptible to making flawed generalizations and repeating errors. This condition indicates that pattern recognition is not merely an additional skill but a fundamental foundation for mathematical information processing.

These findings align with other research that emphasizes the essential role of pattern recognition in cognitive tasks (Lye & Koh, 2014; Shute et al., 2017). Additionally, research has shown that misconceptions in graphs and probability frequently arise from inadequate basic pattern recognition (Sari et al., 2023). However, research has demonstrated that visual manipulatives and interactive media can enhance pattern recognition, highlighting the significance of context-based instructional strategies (Pratiwi & Nashiroh, 2025). Therefore, pedagogical interventions that explicitly focus on identifying patterns are crucial for enhancing students' cognitive tasks (CT) skills.

Abstraction

Students encounter challenges in discerning pertinent information while disregarding extraneous details. In graphical representations, their attention is drawn to striking numerical values rather than general patterns that can serve as the foundation for conclusions. In probability-based problems, they fail to comprehend abstract concepts such as proportions, sample spaces, and theoretical probabilities, thereby hindering their ability to distill information to its core essence. Through open coding, words like vague, unclear, confused, and excessive data were identified during sorting, while selective coding highlighted students' inability to perform conceptual simplification. Axial coding further emphasized that this failure in abstraction results in inaccurate mental models, leading to misguided and misconception-filled problem-solving.

The absence of abstraction has a detrimental impact on subsequent cognitive processing stages, including algorithmic thinking and evaluation, as students lack a clear framework for sequencing logical steps. This causes students to become easily distracted by irrelevant information, renders problem-solving processes unfocused, and results in incorrect answers. Furthermore, the failure to simplify information diminishes their capacity to construct accurate mental models essential for prediction, generalization, or mathematical decision-making.

These findings are consistent with research that underscores the significant influence of experience in selecting key features from data on abstraction ability (Lye & Koh, 2014; Wing, 2006). It is evident that explicit, context-based, and visual practice can substantially enhance this skill. Consequently, developing abstraction skills necessitates systematic pedagogical interventions that facilitate students' focus on essential elements and the construction of accurate mental models, thereby enabling subsequent cognitive processing stages to be executed more effectively (Grover & Pea, 2013).

Algorithmic Thinking

Most students respond to tasks based on guesswork, trial-and-error, or personal logic without adhering to systematic steps. In graph tasks, students fail to analyze data sequentially, and in probability tasks, they neglect to document calculation steps, leading to invalid results. Open coding revealed words such as random, trial-and-error, unordered, did not check, and did not calculate, indicating a lack of procedural thinking patterns. Selective coding categorized these findings as an inability to construct sequential processes, while axial coding emphasized that the absence of systematic thinking results in illogical and difficult-to-justify answers. This condition impedes students' ability to solve problems accurately and systematically.

These findings are supported by other research, which demonstrated that secondary school students often fail to apply algorithmic steps without explicit practice (Grover & Pea, 2013). Problem-solving exercises can significantly enhance algorithmic thinking (Lye & Koh, 2014). Although algorithmic thinking is not always necessary in non-computational tasks, it remains crucial for producing valid and consistent answers in AKM problems (Ashiddiqi et al., 2024). Therefore, developing algorithmic thinking necessitates structured practice and explicit instruction to enable students to follow logical problem-solving steps.

Data Representation

Students encounter difficulties in interpreting relationships between variables in graphs, comprehending trends, and comparing values across data points. In probability problems, they also fail to accurately represent sample spaces and events. Open coding revealed terms such as misread graph, did not see table, did not compare values, and missing table, indicating their inability to transform visual representations into accurate mental representations. Selective coding confirmed that this challenge reflects a fundamental issue in mapping data into numerical or symbolic formats. Axial coding further emphasized that incorrect initial representation impacts the entire reasoning process, resulting in inaccurate final answers.

Students' data representation skills are inadequate across graphs, tables, and probabilistic tasks, leading to imprecise data interpretation and erroneous decisions. This difficulty also affects other cognitive-technical (CT) aspects, as students are unable to construct accurate mental models, structure algorithmic steps, or evaluate their answers logically. These obstacles suggest that data representation is not merely an additional skill but a foundational ability that influences the entire computational thinking process. Emphasizing that graph interpretation and visual data reading are among students' primary weaknesses (Sari et al., 2023). Data representation can be enhanced through multimodal representation, such as combining tables, diagrams, and graphs (Pusat Asesmen Pendidikan, 2024). Therefore, instructional strategies that incorporate various visual representations and intensive practice are essential to assist students in comprehending data and drawing valid conclusions.

Evaluation (Logical Verification)

Students frequently overlook checks, disregard the plausibility of answers, and accept final results without verification. In both graph and probability tasks, students fail to reflect on their results, allowing minor errors to persist into the final answers. Open coding revealed words such as unsure, hesitant, did not check, and illogical, indicating low self-evaluation in problem-solving. Selective coding highlighted that evaluation is one of the weakest aspects of students' critical thinking (CT), while axial coding demonstrated that without evaluation, early mistakes remain undetected and final answers remain incorrect.

Students' evaluation ability is extremely low, so reflective behavior necessary for accurate problem-solving does not emerge. This directly affects their inability to produce valid answers, even when initial data is available. These findings are consistent with Polya (1945) and Schoenfeld (1985), who showed that evaluation is the most frequently neglected step in students' problem-solving. Evaluative ability develops through metacognitive reflection practice (Ashiddiqi et al., 2024). Although some studies argue that evaluation is not always required for simple tasks, these findings confirm that, even in abstract mathematical problems (AKM), evaluation remains a crucial component that cannot be ignored.

CONCLUSIONS

Students' Computational Thinking (CT) skills remain low across all indicators, particularly in decomposition, pattern recognition, abstraction, algorithmic thinking, and evaluation. This results in partial data reading, misinterpretation of graphs, and failure to construct accurate probability representations. Analysis of students' responses to AKM questions using Grounded Theory revealed recurring cognitive obstacles: open coding highlighted frequent expressions such as confused, unsure, misread graph, and no steps, indicating difficulties in systematic information processing. Selective coding categorized these into key CT components, revealing struggles in problem decomposition, pattern recognition, concept simplification, algorithmic step organization, and answer verification. Axial coding revealed interrelated weaknesses, where errors in early stages propagated to later problem-solving steps. The study has limitations, including a small sample size, only two contexts (crop graphs and probability), and qualitative data derived solely from written responses without in-depth interviews. These limitations may limit the interpretation of students' overall thinking processes. Nevertheless, the findings suggest the need for instructional interventions that explicitly integrate CT into mathematics content, utilize scaffolding to support data interpretation, and employ richer assessment instruments to comprehensively capture students' cognitive processes. Grounded Theory can provide valuable insights into the sequence and interconnection of CT difficulties, informing targeted pedagogical strategies.

REFERENCES

Ahmad, A. (2022). Pemantauan pelaksanaan asesmen nasional berbasis komputer (ANBK) sekolah dasar binaan Kecamatan Kopang Kabupaten Lombok Tengah tahun 2021. *Jurnal Paedagogy, 9*(1), 34. https://doi.org/10.33394/jp.v9i1.4619

Angeli, C., & Giannakos, M. (2020). Computational thinking education: Issues and challenges. *Computers in Human Behavior, 105*, 106185. https://doi.org/10.1016/j.chb.2019.106185

Ashiddiqi, H., Sukoriyanto, S., & Hidayah, I. N. (2024). Kesalahan peserta didik dalam menyelesaikan soal AKM pada level kognitif reasoning berdasarkan teori Newman. *FIBONACCI: Jurnal Pendidikan Matematika dan Matematika, 10*(2), 247–260. https://doi.org/10.24853/fbc.10.2.247-260

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

Bioto, A. R., Suking, A., & Zulystiawati, Z. (2022). Analisis kesiapan guru dalam menghadapi asesmen nasional. *Student Journal of Educational Management*, 31–45. https://doi.org/10.37411/sjem.v2i1.1050

Corbin, J. M., & Strauss, A. L. (2008). *Basics of qualitative research: Techniques and procedures for developing grounded theory* (3rd ed.). Sage Publications.

Creswell, J. W. (2014). *Research design: Qualitative, quantitative, and mixed methods approaches* (4th ed.). SAGE Publications.

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

Kementerian Pendidikan dan Kebudayaan. (2020). *AKM dan implikasinya pada pembelajaran*. Pusat Asesmen dan Pembelajaran. https://pusmendik.kemdikbud.go.id/an/file_akm2_202101_1.pdf

Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi. (2021). *Asesmen nasional: Lembar tanya jawab*. Pusat Asesmen dan Pembelajaran, Badan Penelitian dan Pengembangan dan Perbukuan. https://pusmendik.kemdikbud.go.id/an/file_akm_202101_1.pdf

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K–12? *Computers in Human Behavior, 41*, 51–61. https://doi.org/10.1016/j.chb.2014.09.012

OECD. (2019). *PISA 2018 assessment and analytical framework*. OECD Publishing. https://doi.org/10.1787/b25efab8-en

Partnership for 21st Century Learning. (2019). *Framework for 21st century learning definitions*. Battelle for Kids. https://www.battelleforkids.org/networks/p21

Polya, G. (1945). *How to solve it: A new aspect of mathematical method*. Princeton University Press.

Pratiwi, E., & Nashiroh, P. K. (2025). Meta-analysis pengaruh media interaktif terhadap kemampuan berpikir komputasional dalam pembelajaran pemrograman. *Jurnal Riset dan Inovasi Pembelajaran, 5*(2), 692–710. https://doi.org/10.51574/jrip.v5i2.3705

Pusat Asesmen Pendidikan. (2024). *Profil satuan pendidikan dengan capaian AKM tinggi pada jenjang SMA/SMK/MA/sederajat: Dokumen rekomendasi bahan kebijakan hasil asesmen nasional tahun 2023*. Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi.

Pusat Penilaian Pendidikan. (2020). *Desain pengembangan soal asesmen kompetensi minimum*. Kementerian Pendidikan dan Kebudayaan. https://pusmendik.kemdikbud.go.id/an/page/asesmen_kompetensi_minimum

Rahmawati, S. N., Fauzi, A., & Supurwoko, S. (2024). Computational-thinking based learning activities for physics in Indonesian new curriculum through a student worksheet development. *JIPF (Jurnal Ilmu Pendidikan Fisika), 9*(3), 426. https://doi.org/10.26737/jipf.v9i3.4807

Rochmaeni, F. S. K., & Wardana, M. D. K. (2023). Student's difficulties in completing questions of numeracy literacy minimum competency assessment: Kesulitan siswa dalam menyelesaikan soal asesmen kompetensi minimum literasi numerasi. https://doi.org/10.21070/ups.2489

Sari, M. C. P., Mahmudi, M., Kristinawati, K., & Mampouw, H. L. (2023). Peningkatan kemampuan representasi matematis melalui model problem based learning. *PTK: Jurnal Tindakan Kelas, 4*(1), 1–17. https://doi.org/10.53624/ptk.v4i1.242

Schoenfeld, A. H. (1985). *Mathematical problem solving*. Academic Press.

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

Wijaya, A., Van den Heuvel-Panhuizen, M., & Doorman, M. (2015). Opportunity-to-learn context-based tasks provided by mathematics textbooks. *Educational Studies in Mathematics, 89*(1), 41–65. https://doi.org/10.1007/s10649-015-9595-1

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