25 (4), 2024, 1752-1770

Jurnal Pendidikan MIPA

e-ISSN: 2685-5488 | p-ISSN: 1411-2531

http://jurnal.fkip.unila.ac.id/index.php/jpmipa/

Knowledge to be Taught of Computational Thinking: A Praxeological Analysis

Nusuki Syari'ati Fathimah, Andini Setya Arianti, & Enjun Junaeti*

Computer Science Education Study Program, Universitas Pendidikan Indonesia, Indonesia

Abstract: Efforts to enhance the quality of informatics education at the secondary school level within the Merdeka Curriculum focus on identifying learning barriers. This study analyzes secondary school informatics textbooks to identify potential learning obstacles through praxeological analysis. The analysis focuses on tasks, techniques, technologies, and theories employed in the textbooks to determine whether the construction of students' knowledge is epistemically aligned. Based on the praxeological analysis, 20 tasks were identified, categorized into five task types. The techniques used to solve these tasks are aligned with computational thinking skills. Four reference theories are consistent with the data and algorithm analysis elements integrated with computational thinking elements in the latest curriculum: algorithms, data representation, data structures, and optimization. This study identifies several potential learning barriers: ontogenic obstacles arise from the sequencing of task types, epistemological obstacles from the pre-existing conceptual frameworks, and didactic obstacles from the clarity of instructions and task complexity. In conclusion, this study recommends improvements to the reference epistemological model, particularly in adjusting instructions and the sequencing of tasks.

Keywords: praxeology, knowledge to be taught, informatics, learning obstacle, computational thinking.

▪ INTRODUCTION

Informatics at the middle school level in Indonesia was first introduced with eight elements (Wijanto et al., 2021), including Computational Thinking (CT), Algorithms and Programming (AP), Data Analysis (DA), Computer Networks and the Internet (CNI), Computer Systems (CS), Information and Communication Technology (ICT), Social Impacts of Informatics (SII), and Cross-Disciplinary Practices (CDP), marking a distinctive feature of the Merdeka Curriculum implementation. Despite recent government policies reducing these elements to only two, CT and Digital Literacy (DL), it is undeniable that informatics education at the elementary level (both primary and middle school) plays a crucial role in shaping students' understanding of basic information technology concepts (Nabilah et al., 2023).

The constructivist approach underlying the Merdeka Curriculum emphasizes learning that actively involves students in building their own knowledge (Cholilah et al., 2023; Dwiputra et al., 2023; Hakiky et al., 2023; Juliangkary et al., 2023; Nazihah & Nurcahyo, 2019; Pata'dungan et al., 2023). Ideally, teachers should personalize the curriculum (knowledge to be taught) and contextualize it to produce teachable materials (taught knowledge) as part of a didactic transposition (Chevallard & Bosch, 2020). Therefore, the teacher's role is crucial in creating learning environments that promote active student knowledge construction.

However, the implementation of informatics teaching practices in the field is often not carried out by teachers with professional qualifications in informatics. This issue pertains to the Minister of Education, Culture, Research, and Technology Decree No.

Enjun Junaeti
*Email: enjun@upi.edu

DOI: http://dx.doi.org/10.23960/jpmipa/v25i4.pp1752-1770
Received: 31 December 2024
Accepted: 13 January 2025
Published: 16 January 2025

262/M/2022, which provides guidelines for curriculum implementation in learning recovery, and the Education Curriculum and Assessment Standards Agency Decree No. 033/H/KR/2022, which outlines learning outcomes for middle school informatics subjects within the Merdeka Curriculum. These documents state that informatics teachers must have a minimum educational background of a bachelor's degree (S-1/D-IV) in relevant disciplines, such as computer science/ informatics, mathematics, and natural sciences.

Professional competence ensures that teachers can personalize and contextualize the curriculum effectively (Pansell, 2023). Although efforts to provide technical guidance to Informatics teachers at various levels are ongoing, conducted both by the government and independently by several universities, the implementation often coincides with the Merdeka Curriculum rollout. Consequently, many teachers tend to rely on middle school informatics textbooks as the primary source of taught knowledge. On the other hand, inappropriate didactic design can lead to learning obstacles (Brousseau et al., 2002), affecting students' formation of concept images (Suryadi, 2019).

This study aims to identify learning obstacles in middle school informatics textbooks, specifically in the context of computational thinking (CT). By applying praxeological analysis, the research will examine how the textbooks present CT concepts and whether they align with effective pedagogical practices. The objective is to identify any potential didactic obstacles that may hinder students' ability to understand and internalize CT concepts. Praxeological analysis will focus on the tasks (T), techniques (τ), and the theoretical (Θ) and technological (θ) foundations of the knowledge presented in the textbooks (Chevallard, 2007). This approach will help identify obstacles—didactic, epistemological, and ontogenic—that may arise from misalignments between the tasks, techniques, and students' cognitive readiness or developmental stages. These obstacles may impede students' learning within the Zone of Proximal Development (ZPD) (Sidik et al., 2021).

Several studies have explored the application of praxeological analysis within educational settings. Takeuchi and Shino (2020) applied praxeology to compare the content of mathematics textbooks from Japan and England. Utami et al. (2024) examined the approach used by Indonesian textbooks in presenting the concept of functions at the lower secondary school level. Yunianta et al. (2023) conducted an analysis of curriculum materials in mathematics textbooks, focusing on the measurement of spatial figures. Putra et al. (2021) investigated the mathematical and didactic competencies reflected in mathematical challenges of comics book designed by preservice elementary teachers. Wijayanti and Winsløw (2017) provided a quantitative “profile” of textbooks in the context of arithmetic proportions. Rønning (2022) explored the connection between mathematics and its applications in engineering. Despite the considerable body of research on praxeological analysis in various educational contexts, its application within informatics education—especially concerning Indonesian middle school textbooks—remains relatively underexplored. This study, therefore, extends the existing body of literature by applying praxeological analysis to evaluate the presentation of computational thinking (CT) in middle school informatics textbooks and to identify potential obstacles to effective learning. Incorporating additional relevant studies would further strengthen the framework for understanding the essential role of textbook design in influencing students' learning experiences.

The scope of this research is specifically focused on middle school informatics textbooks used in Indonesian schools under the Merdeka Curriculum. The selected textbooks will be analyzed based on criteria such as their alignment with curriculum guidelines, the quality of the tasks and techniques presented, and the pedagogical approaches employed. By focusing on these elements, the study will explore how the textbooks may either facilitate or obstruct students' engagement with CT, with the aim of offering recommendations to improve the alignment of textbook content with students' cognitive and developmental needs.

In conclusion, this research aims to identify learning obstacles in the presentation of computational thinking in middle school informatics textbooks through praxeological analysis. The findings will help inform how these textbooks can be better designed to promote effective learning and enable students to fully engage with computational thinking concepts. Ultimately, this study seeks to contribute to improving informatics education in Indonesia by addressing the challenges faced by both teachers and students in the context of the Merdeka Curriculum.

▪ METHOD

Participants

This research exclusively examined educational materials rather than involving human subjects, such as teachers or students. The primary focus was on Grade 7 middle school informatics textbooks utilized in Indonesia under the Merdeka Curriculum framework (Wijanto et al., 2021). Additional sources included teacher support materials for Grade 7 informatics, published by the Center for Books of the Education Standards, Curriculum, and Assessment Agency of the Ministry of Education, Culture, Research, and Technology (Wisnubhadra et al., 2021). These textbooks and curriculum documents were chosen due to their representative nature in conveying computational thinking and other informatics concepts at the middle school level. Furthermore, supplementary secondary sources consisted of Ministerial Decrees (No. 262/M/2022 and No. 033/H/KR/2022) related to informatics education, as well as referenced scholarly works (Cormen et al., 2022) and relevant academic articles, including those by Basu et al. (2016), Hsu et al. (2018), Lamprou and Repenning (2018), Lockwood and Mooney (2018), and Lye and Koh (2014).

Research Design and Procedures

This study employed a qualitative content analysis design to examine the praxeological organization of knowledge in the sampled textbooks. The research began with the selection of relevant materials during the first three weeks of the first month, where Grade 7 textbooks and secondary sources were identified and collected based on their use in classrooms and alignment with the Merdeka Curriculum. In the fourth week of the first month, the praxeological framework, adapted from Chevallard (2007), Takeuchi & Shinno (2020), and Yunianta et al. (2023), was refined to suit the context of middle school informatics education. This framework, illustrated in Figure 1, comprises four key elements tasks, techniques, technology, and theory—which were analyzed in relation to the praxis (know-how) and logos (knowledge) dimensions, serving as the foundation for subsequent analyses.

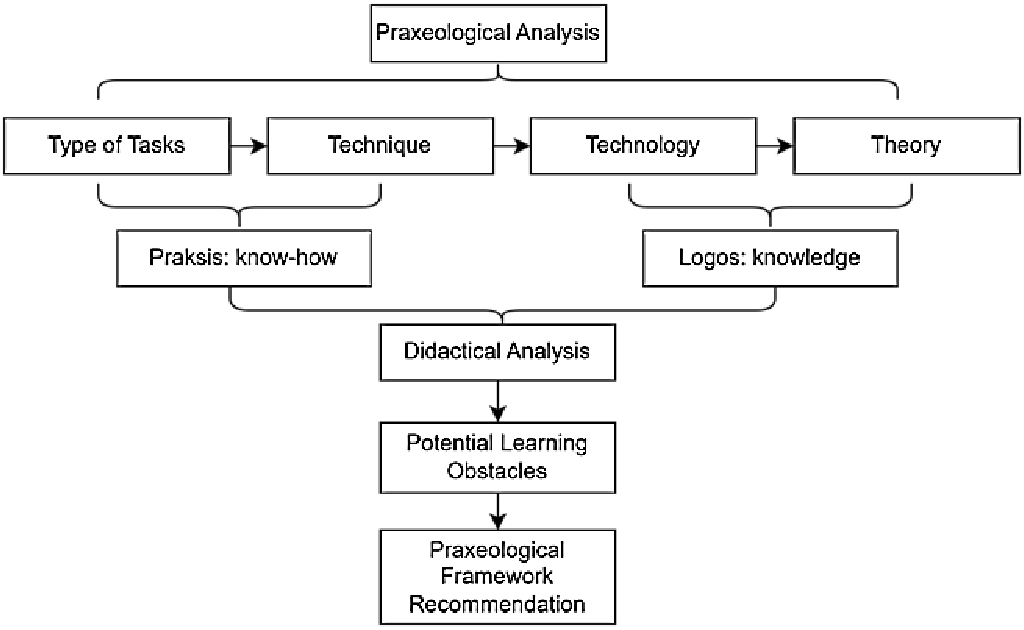


Figure 1. Research stages of praxeological analysis

During the second month, the content of the textbooks was systematically coded to identify and categorize praxeological elements, focusing on themes and patterns related to computational thinking. Subsequently, the coded data were analyzed didactically to uncover potential ontogenic, epistemological, and didactic learning obstacles that might hinder student understanding, employing iterative processes as described by Yin (2016) to ensure rigorous identification of obstacles. Finally, during the third month, validation and triangulation were conducted by cross-referencing data from textbooks, teacher support materials, and curriculum documents. This process adhered to methodological triangulation guidelines provided by Creswell & Creswell (2017), ensuring accuracy and consistency of findings.

Instruments

The primary instrument for this study was a content analysis coding framework, adapted from Chevallard's praxeological model (Chevallard, 2007) and refined with insights from Takeuchi & Shinno (2020). This framework was validated through an expert review conducted by three subject-matter experts specializing in informatics education and computational thinking, consisting of two education experts and an experienced practitioner teacher. Reliability was ensured by conducting iterative discussions and resolution of discrepancies among coders during the coding process.

Data Analysis

The data were analyzed using qualitative content analysis techniques as described by Elo & Kyngäs (2008). Coding was performed to extract praxeological elements and themes related to computational thinking. The types of tasks were coded according to the cognitive actions required to solve the problem as adapted from Polat et al. (2022), while the techniques were analyzed in terms of the computational thinking elements they aimed to promote. Additionally, the logos component comprising theory and technology referred to informatics concepts introduced, specifically algorithms and data analysis, along with the rules associated with these concepts to justify the validity of the techniques employed. Thematic analysis was used to identify patterns and potential learning obstacles. Triangulation was conducted by comparing data across textbooks, teacher support materials, and curriculum documents to enhance validity and reliability. Didactical analysis was conducted to identify potential learning obstacles and provide recommendations for improving the praxeological structure of the teaching materials. These findings provide insights into the alignment of the materials with the intended learning outcomes of the Merdeka Curriculum.

▪ RESULT AND DISSCUSSION

This section presents the study's findings. The textbook's organization of computational thinking (CT) elements is analyzed to examine how key concepts are structured and integrated into the learning materials. The Praksis Block is reviewed to evaluate the alignment of tasks and techniques with CT principles. The Logos Block is analyzed to ensure coherence between tasks and the informatics concepts taught. References to epistemological models are used to assess the knowledge frameworks in the textbook and their theoretical soundness. A didactic analysis evaluates the pedagogical approaches and their effectiveness in supporting student learning, identifying potential obstacles in presenting CT concepts at a cognitive level appropriate for middle school students. Recommendations are provided to address these obstacles, guided by epistemological models for improving instructional design and textbook development.

Organization of the Textbook for CT Element

The Grade 7 Informatics textbooks for the computational thinking (CT) component are structured around five primary tasks that students are expected to complete. Table 1 illustrates the textbook's organization for the CT element. The content structure is clearly outlined through well-organized subchapters, including Introduction, Algorithm, Scheduling Optimization, Data Structure, Data Representation, and Competency Test. Each subchapter emphasizes specific concepts, allowing readers to follow the logical progression of ideas with ease. This arrangement not only provides a clear framework but also enables students to grasp the interconnections between concepts and systematically delve into informatics topics. The inclusion of data structures further aligns with the curriculum areas currently taught in schools in England and Lithuania (Dagienė & Sentance, 2016).

Table 1. Structure of textbook for the computational thinking element

Subchapter	Concept Delivered	Context	Informatics Concept Desired
Introduction: Computational Thinking	Computational thinking involves solving problems in a way that can be executed by a computer.	The difference between homemade cookies and packaged biscuits sold in stores.	Computational Thinking

Subchapter	Concept Delivered	Context	Informatics Concept Desired
Algorithm	Patterns or rules for performing various activities.	(1) Rules for students to be allowed to take exams. (2) Rules for participants in sports competitions. (3) Arranging beads to make a bracelet.	Algorithm is a set of instructions for solving a problem.
	Effective and efficient processes.	Biscuit production.	
Scheduling Optimization	Scheduling activities for specific times and managing parallel tasks.	(1) Daily activity scheduling. (2) Filling a bucket with water.	Scheduling algorithms
Data Structure	Data organized in a list, often called a linked list in Informatics.	(1) Shopping lists. (2) Class rosters. (3) Creating a password.	Data in a linked list
Data Representation	Binary choices: yes or no.	(1) Whether breakfast was eaten. (2) Chances of rain. (3) Favorite color being blue. (4) Classroom booking schedules.	Binary data and OR operations on binary numbers.
Test	-	Sharing wooden sticks.	-

When linked to the learning objectives related to utilizing computational thinking for problem-solving, the content structure effectively supports the achievement of these goals. Moreover, this organization enhances conceptual understanding and promotes practical application in real-world scenarios. However, when considering the difficulty levels, certain tasks, such as arranging beads to make a bracelet and creating a password, involve complex problem-solving, which contrasts with findings from Dagiene & Dolgopolovas (2022) suggesting that short tasks are more suitable for scaffolding computational thinking.

The teacher's book includes enrichment and remedial sections with problem-solving references from Bebras challenge questions and provides guidance on related informatics concepts. Tabel 2 and 3 show context and informatics concept desired. The enrichment tasks are generally well-aligned with the informatics concepts, though some contexts may require further clarification or adjustments to enhance understanding.

Table 2. Enrichment tasks

Task	Context	Informatics Concept Desired
1	Arranging bead bracelets.	Algorithm
2	Filling a bucket with water.	Scheduling algorithms
3	Creating a password.	Linked list
4	Classroom booking schedules.	Binary data

5	Connecting circles.	Step efficiency.
6	Three otters in a box.	Sorting networks.
7	Three in a row.	Algorithms and state transitions.
8	Musical chairs.	Algorithms and patterns.
9	Candy collecting robot.	Brute force and dynamic programming recurrence.
10	Walking in the park.	Graph data structures.

The proposed remedial tasks are largely appropriate for Grade 7 students, focusing on relevant informatics concepts. Each task encourages critical thinking and practical application of the concepts being taught.

Table 3. Remedial tasks

Task	Context	Informatics Concept Desired
1	Ninja names.	Coding, cryptography, and patterns.
2	Packing apples.	Binary numbers.
3	Donut queue.	Scheduling.
4	Honomakato bridge.	Graph data structures.

Praksis Block: Type of Task and Technique

The textbooks selected for analysis emphasize tasks aimed at developing students' computational thinking skills through problem-solving scenarios involving small-scale discrete data. A total of 20 tasks were extracted from the student and teacher books and analyzed using praxeological analysis. The tasks were subsequently grouped into five categories according to the cognitive actions required to solve the problem, as adapted from Polat et al. (2022): Object Arrangement (T_1), Task-completion (T_2), Enumeration (T_3), Networking (T_4), and State Transition (T_5). Each type of task is further divided into subtasks, which are organized according to progressively increasing levels of complexity and the specific logical operations required for each step. The identification of techniques (τ) used to address these tasks reveals the computational thinking strategies being cultivated. Specifically, these strategies include abstraction (Ab - τ_1), Decomposition (D - τ_2), Pattern Recognition (P - τ_3), and Algorithmic Thinking (Ag - τ_4), as outlined in Wijanto et al. (2021) and Wisnubhadra et al. (2021). Table 4 presents a comprehensive overview of these task categories and the corresponding techniques required for each.

Object Arrangement tasks (Nagata & Nishi, 2021) involve organizing or arranging elements according to specific rules or patterns. Solving these tasks necessitates logical reasoning, as students must determine appropriate positions, sequences, or configurations. There are three main tasks t_1, t_2, t_7, three enrichment tasks t_7, t_9, t_{12}, and one remedial task t_{19} categorized under this type. However, the occurrence of these tasks does not align with progressively increasing levels of complexity and the specific logical operations required. A similar issue arises with Networking tasks (Camacho et al., 2020), which require students to analyze connections and map relationships between nodes or points, such as elements within a graph or network. This cognitive action entails understanding connections and finding optimal paths or configurations. In this case, the questions focusing on searching and optimizing elements are presented before those centered on identifying network paths. This misalignment in task presentation can potentially lead to learning obstacles.

Table 4. Praksis block for the computational thinking element

Types of Tasks	Subtypes of Task	Examples of Task	Techniques
T_1: Object Arrangement	$T_{1,1}$: Arranging Objects	Secret Word (t_4, t_9) Ninja Name (t_{17})	P. Ab D. P. Ag
	$T_{1,2}$: Counting Arranged Objects Length	Arranging bead bracelets (t_1, t_2, t_7)	Ag. P. Ab
	$T_{1,3}$: Calculating Arrangement Steps	Three Otters in a Box (t_{12})	Ag. Ab
T_2: Task-completion	$T_{2,1}$: Optimizing Task Completion Time	Donut Queue (t_{19})	D. P. Ag. Ab
	$T_{2,2}$: Measuring Task Completion Time	Filling a Bucket with Water (t_3, t_8)	Ag. Ab
T_3: Enumeration	$T_{3,1}$: Counting Elements with Conditions	Room Booking (t_5, t_{10}) Sharing Wooden Sticks (t_6) Packing Apples (t_{18})	Ab D P. Ag. Ab
T_4: Networking	$T_{4,1}$: Network Mapping	Connecting Circles (t_{11}) Honomakato Bridge (t_{20})	Ag. Ab D. P. Ag. Ab
	$T_{4,2}$: Identifying Network Paths	Walking in the Park (t_{16})	D. Ag. Ab
	$T_{4,3}$: Searching and Optimizing Elements	Candy Collector Robot (t_{15})	D. P. Ag. Ab
T_5: State Transition	$T_{5,1}$: Identifying Initial State with Conditions	Three in a Row (t_{13}) Musical Chair (t_{14})	D. Ab D. P. Ag. Ab

Completion tasks (Leova et al., 2022) require students to complete a series of actions to achieve a defined goal. This type involves sequential planning, where each step must be executed in the correct order to successfully accomplish the task, emphasizing the cognitive actions of planning and execution. Meanwhile Enumeration tasks focus on counting or listing items based on established criteria. Students utilize quantitative reasoning to tally up or identify all possible combinations that meet certain conditions, enhancing their counting skills. Lastly, State Transition tasks necessitate tracking or predicting changes in a system's state based on specific actions or rules. Students must manage different states and comprehend how various operations or inputs impact the system's status, reinforcing their understanding of state management. The presence of these three task types corresponds with progressively increasing levels of complexity and the particular logical operations required.

The techniques employed across tasks exhibit varying frequencies of computational thinking elements, with the technique of decomposition being less prominent in the primary tasks. This, however, is justifiable, as the problems presented do not necessitate extensive decomposition into smaller components, given the target audience of middle school students. Recent studies (Jiang & Li, 2021; Pan et al., 2024; Rich et al., 2019) affirm the significance of these techniques in fostering computational thinking skills among middle school students, highlighting their relevance within educational contexts. This is further supported by research from Delal & Oner (2020), which provides valuable insights into how computational thinking elements are effectively integrated into middle school curricula to enhance students' problem-solving abilities. However, Bati et al. (2018) contend that an excessive focus on these elements could overshadow practical, hands-on problem-solving skills that are equally essential in real-world contexts. They argue that curricula should strike a balance between abstract reasoning and practical applications, encouraging students to engage in interdisciplinary projects that integrate computational thinking with other fields such as biology, engineering, and the social sciences.

Logos Block: Technology and Theory

Table 5 presents the Logos Block to support the development of computational thinking. It aligns theoretical constructs with specific sub-theories, their technological applications, and task examples that illustrate their use. The theories referenced are selected based on informatics element to be integrated into the computational thinking, aligned with the learning outcomes in the Phase D informatics curriculum, specifically focusing on data analysis and algorithms. The sequence is adjusted based on one of the referenced scholarly knowledge sources (Cormen et al., 2022). The technologies used are derived from informatics concepts that align with the selected theories.

Table 5. Logos block for the computational thinking element

Theory	Sub theory	Technology	Examples of Task
Θ_1: Algorithm	$\Theta_{1,1}$: Iterative	$\theta_{1,1}$: Greedy Choice Property $\theta_{1,2}$: Optimal Substructure $\theta_{1,3}$: Backtracking Algorithm $\theta_{1,4}$: Logic $\theta_{1,5}$: Substitution	Arranging bead bracelets (t_1, t_2, t_7) Three in a Row (t_{13}) Musical Chair (t_{14}) Ninja Name (t_{17})
	$\Theta_{1,2}$: Dynamic Programming	$\theta_{1,6}$: Max-flow	Candy Collector Robot (t_{15})
Θ_2: Data Representation	$\Theta_{2,1}$: Binary Data	$\theta_{2,1}$: OR Operation $\theta_{2,2}$: Remainder Principle	Room Booking (t_5, t_{10}) Packing Apples (t_{18})
Θ_3: Data Structure	$\Theta_{3,1}$: List	$\theta_{3,1}$: Linked list	Secret Word (t_4, t_9)
	$\Theta_{3,2}$: Graph	$\theta_{3,2}$: Sorting Principle $\theta_{3,3}$: Connectivity $\theta_{3,4}$: Path Labelling $\theta_{3,5}$: Adjacency	Three Otters in a Box (t_{12}) Honomakato Bridge (t_{20}) Walking in the Park (t_{16}) Connecting Circles (t_{11})
Θ_4: Optimization	$\Theta_{4,1}$: Schedulling $\Theta_{4,2}$: Resource Allocation	$\theta_{4,1}$: Parallel Work $\theta_{4,2}$: Queque (FIFO) $\theta_{4,3}$: Knapsack Theorem	Filling a Bucket (t_3, t_8) Donut Queue (t_{19}) Sharing Wooden Sticks (t_6)

Middle school students are in the transition from the concrete operational to the formal operational stage of cognitive development. At this age, they can begin to understand abstract ideas but benefit most from concepts grounded in visual, tangible, or gamified representations. The variety of tasks and technologies outlined in the table offers a rich foundation for teaching CT concepts to middle school students, but their implementation must be carefully tailored. Iterative algorithms, basic binary data, and simple graph concepts are particularly suitable when connected to real-life contexts or hands-on activities. For instance, optimization tasks like *Filling a Bucket with Water or Donut Queue* resonate with students when framed around tangible experiences. Conversely, complex techniques such as dynamic programming or advanced graph principles are less appropriate unless significantly simplified, as they might overwhelm students' developing abstract reasoning skills. Overall, adapting the tasks to students' cognitive levels and prior knowledge ensures that they can build CT skills effectively while maintaining interest and understanding.

References of Epistemological Model

Referring to the results of the praxeological analysis conducted, the reference to the epistemological model of computational thinking elements in the student textbook is presented in Figure 2. As noted by Chevallard et al. (2022), references to the epistemological model explicitly encompass specific activities that can be considered the raison d'être of the content involved. If the raison d'être in question pertains to the sequential order of theories, it is recommended that students first understand data representation before proceeding to learn how to efficiently organize data using data structures. While this sequence may not result in immediate didactic obstacles, as the tasks are not interdependent, it is evident from an epistemological perspective that such an order could lead to future challenges (Brousseau et al., 2002).

$$\left.\begin{array}{c}[T_{1,2}/\tau_3,\tau_4/\theta_{1,1},\theta_{1,2}/\Theta_{1,1}]\\ \downarrow\\ [T_{1,2}/\tau_1/\theta_{1,1},\theta_{1,2}/\Theta_{1,1}]\end{array}\right\}\rightarrow [T_{2,2}/\tau_1,\tau_4/\theta_{4,1}/\Theta_{4,1}] \rightarrow \begin{array}{c}[T_{1,1}/\tau_1,\tau_3/\theta_{3,1}/\Theta_{3,1}]\\ \downarrow\\ [T_{2,2}/\tau_1/\theta_{2,1}/\Theta_{2,1}]\\ \downarrow\\ [T_{3,1}/\tau_2/\theta_{4,3}/\Theta_{4,2}]\end{array}$$

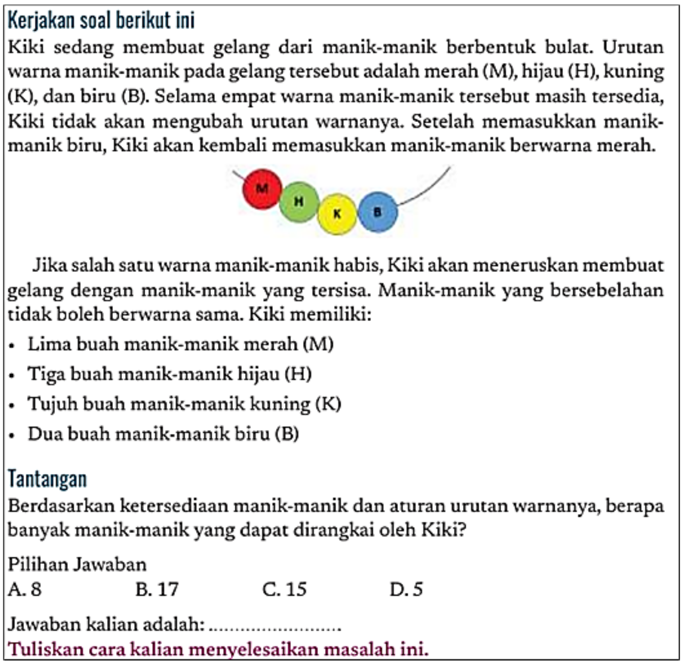
Figure 2. References of epistemological model computational thinking element

Moreover, based on the principle of progressively increasing levels of complexity in problem-solving, the task type *Arranging Objects* should logically precede the task type *Counting Arranged Objects Length*. Insufficient mastery of object arrangement by students may lead to ontogenic obstacles when engaging with *Counting Arranged Objects Length* tasks, as this transition entails a substantial increase in cognitive complexity (Brousseau et al., 2002).

Didactic Analysis

The student tasks are diverse and hands-on, aiming to engage students in various problem-solving practical and cognitive tasks. The tasks cover a range of topics and skills, from sorting principles to max-flow algorithms, ensuring students encounter diverse problem-solving scenarios. This approach aligns with recent findings by (Pérez, 2018), who emphasize the importance of integrating mathematical principles and computational thinking to enhance student engagement and learning outcomes. Tasks consistently incorporate elements such as pattern recognition, decomposition, and abstraction, and fostering critical thinking and algorithmic reasoning. However, the task designed for practice and enrichment are only available in the teacher's book, making the completion of these tasks highly dependent on how the teacher designs the taught knowledge.

Figure 3 shows one of the tasks in the student book that needs to be completed. Since this activity is designed to apply computational thinking to efficiently solve problems involving algorithms," it is recommended to include specific instructional guidance on how to apply computational thinking in problem-solving. For example, in the decomposition strategy, questions like "What information is known?" and "What is the task's objective?" could be added. This can provide students with the opportunity to use the decomposition strategy to distinguish between informational and goal-oriented statements (Beckers et al., 2013).


Kerjakan soal berikut ini

Kiki sedang membuat gelang dari manik-manik berbentuk bulat. Urutan warna manik-manik pada gelang tersebut adalah merah (M), hijau (H), kuning (K), dan biru (B). Selama empat warna manik-manik tersebut masih tersedia, Kiki tidak akan mengubah urutan warnanya. Setelah memasukkan manik-manik biru, Kiki akan kembali memasukkan manik-manik berwarna merah.

M H K B

Jika salah satu warna manik-manik habis, Kiki akan meneruskan membuat gelang dengan manik-manik yang tersisa. Manik-manik yang bersebelahan tidak boleh berwarna sama. Kiki memiliki:

- Lima buah manik-manik merah (M)
- Tiga buah manik-manik hijau (H)
- Tujuh buah manik-manik kuning (K)
- Dua buah manik-manik biru (B)

Tantangan

Berdasarkan ketersediaan manik-manik dan aturan urutan warnanya, berapa banyak manik-manik yang dapat dirangkai oleh Kiki?

Pilihan Jawaban

A. 8 B. 17 C. 15 D. 5

Jawaban kalian adalah: ....

Tuliskan cara kalian menyelesaikan masalah ini.

Figure 3. Example of task in student book

The knowledge required by students to engage in these tasks spans several domains, including mathematical principles, computer science theories, and logical operations. This multi-faceted approach ensures that students develop a strong foundational understanding while also learning to apply this knowledge in various contexts. As stated in (Anderson, 1982), this integration of multiple domains supports comprehensive cognitive development and skill acquisition. However, if there is a domain of knowledge that the student does not master, this will become an ontogenetic conceptual learning obstacle (Suryadi, 2019).

Teacher instructions play a critical role in guiding students through these tasks, providing the necessary scaffolding to ensure comprehension and successful task completion (Sidik et al., 2021). This view is supported by recent research (van de Pol et al., 2019), who found that effective teacher guidance and scaffolding techniques are essential in supporting student learning during practical tasks. Teachers introduce the tasks and provide initial guidance, allowing students to explore and solve problems independently or in groups. Instructions are tailored to the specific tasks, ensuring that students understand the objectives and the methods required to achieve them. Teachers offer feedback and support throughout the activities, helping students to overcome challenges and deepen their understanding. The absence of tailored instructions for specific tasks in the student book may hinder students from developing the desired computational thinking skills due to the lack of instructional scaffolding, potentially leading to didactic learning obstacles.

In the context of education in Indonesia, the class size, which typically ranges from 20 to 35 students, presents a challenge in implementing problem-solving-based learning, particularly in regions with unequal access to technology (Hudha et al, 2023). This learning approach can serve as a solution to enhance critical thinking skills without relying heavily on technological infrastructure. However, it requires clear guidance, especially in the early stages of learning (e.g., in grade 7), where most students are not yet accustomed to problem-solving methods.

Additionally, the limited number of teachers with a background in informatics and the lack of experience in guiding students in competitions like Bebras pose significant challenges. Teachers may struggle to understand and provide solutions for competition-style problems, further complicating the learning process (Lehtimäki et al., 2022). To address the challenges, a scaffolding approach is essential. This method involves starting with simpler problems and gradually increasing complexity, thereby preparing students for competition-level tasks.

The integration of computational thinking into the curriculum remains limited in Indonesia, and there is a pressing need for specialized training to help teachers understand informatics concepts and effective teaching strategies. Cultural factors in learning also play a role, as students are often accustomed to passive learning methods. Transitioning to active, problem-solving-based approaches may require additional time and effort.

Potential Learning Obstacles

Based on the didactic analysis, several potential learning obstacles may hinder students' understanding of computational thinking and related concepts. Ontogenic obstacles are linked to the developmental stage and individual characteristics of learners. Students at various developmental stages may find abstract concepts such as graph theory challenging (González et al., 2021). Referring to Table 3, the Honomakato Bridge task, categorized as a Networking task (Table 4) based on Graph Data Structure theory (Table 5), is recommended as a remedial assignment. However, this recommendation is less appropriate, as students assigned remedial tasks are required to learn new concepts, even though they have not yet mastered the previous ones.

Students with limited prior exposure may struggle to comprehend complex problems, such as those presented in the Counting Arranged Objects Length task, especially without sufficient mastery of Object Arrangement. As illustrated in the task sequence outlined in the References of the Epistemological Model in Figure 2, these types of tasks appear in a sequence where more complex tasks are introduced earlier. In contrast, (Angeli & Valanides, 2009) emphasize that a lack of foundational knowledge significantly hinders the construction of students' knowledge. These studies align with the observation that students who have not mastered basic principles may find it difficult to justify the solutions, which will also lead to epistemological learning obstacles (Brousseau et al., 2002). Additionally, these potential obstacles in the task sequence are related to the theory of data representation and data structures, as evidenced by the disordered sequence. These obstacles are linked to the raison d'être of the sequential ordering of theories.

Didactic obstacles arise from teaching methods and the educational environment (Brousseau et al., 2002). As shown in Figure 3, the problem-solving tasks are introduced without any guidance. Similarly, other tasks follow the same approach, as the problem-solving exercises are directly taken from Bebras competition questions without adaptation to the general conditions of middle school students. Furthermore, the majority of teachers, who often lack a background in informatics, may also be unprepared to address problems resembling competition-style questions like those in Bebras. Some students may need more time and different methods to grasp the same concepts, leading to disparities in learning outcomes. If teacher instructions are unclear or there is insufficient scaffolding, students might struggle to understand the tasks and the concepts they are supposed to learn (Sidik et al., 2021). This issue is underscored by (Vermunt, 2006), who highlight the need for teachers to balance guiding students and allowing them to explore independently. In contrast, Reiser (2018) point out that some tasks might be too complex or not appropriately scaffolded for the students' current level of understanding, which can overwhelm students and discourage them from fully engaging with the activities. This represents a psychological ontogenic obstacle (Suryadi, 2019). Finally, the availability and effective use of resources such as tools, materials, and technological aids are crucial. Inadequate resources or improper use can hinder the learning process, especially for hands-on activities that require specific tools (Asheela et al., 2020). However, this contrasts with the view that learning obstacles arising from educational tools are considered instrumental ontogenic obstacles, as noted by Suryadi (2019).

Praxeological Framework Recomendations

Based on the praxeological analysis and considering the identified learning obstacles, here are refined recommendations for a praxeological framework to address these challenges effectively:

1. Personalized Learning Pathways to Address Didactic Obstacles: Adapt instructional methods to suit the varied learning styles and paces of students at different stages of development (Reiser, 2018; Tsai et al., 2021; van de Pol et al., 2019). Implement differentiated learning paths that provide foundational modules and scaffolding to support students' comprehension of abstract computational concepts. Introduce differentiated learning pathways that include foundational modules and scaffolded support to enhance students' understanding of complex computational concepts. For example, students can be grouped based on their problem-solving abilities, with guiding questions provided for those who are less familiar with problem-solving, such as asking what is already known and what needs to be solved or offering simpler problem to begin with.
2. Rearranging the Task Sequence to Address Epistemological and Ontogenic Obstacles: The arrangement of tasks should be based on the raison d'être of the involved informatics theory in the scholarly knowledge to reduce epistemological obstacles.

Additionally, the progressively increasing levels of complexity of each task type should be considered to prevent the emergence of ontogenic obstacles. Some minor adjustments of Reference Epistemological Model for Computational Thinking tasks are shown in Figure 4.

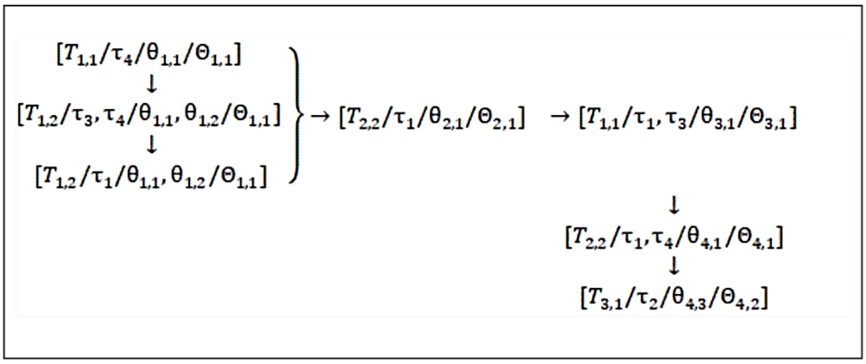


Figure 4. References of epistemological model recomendation

As required by the raison d'être of the involved informatics theory, the tasks are arranged in the sequence to $\Theta_1, \Theta_2, \Theta_3$ and Θ_4 to prevent epistemological obstacles. The addition of the task $T_{1,1}$ in Θ_1 is related to sequences of the *Object Arrangement* (T_1) type of task to reduce ontogenic obstacles.

▪ CONCLUSION

The praxeological analysis of 20 tasks in the textbooks categorizes them into five types, each progressively building on fundamental computational thinking skills. Object Arrangement tasks lay the foundation for pattern recognition and sequencing, followed by Task-completion tasks introducing efficiency and optimization. Enumerating tasks involve set theory and classification, while Networking tasks introduce graph theory and optimization algorithms. State Transition tasks require managing changes within a system, combining prior skills into higher-level logical reasoning. Potential learning obstacles are identified from a didactic perspective, including ontogenic, epistemological, and didactic barriers. Ontogenic obstacles relate to students' developmental stages, where younger learners might struggle with abstract concepts as well as type of task sequences. Epistemological obstacles involve the nature of the knowledge and students' preconceptions, which can hinder the acceptance and understanding of new information. Didactic obstacles pertain to teaching methods and materials, requiring clarity, gradual progression, and relatable contexts to prevent overwhelming students. To overcome learning obstacles, teachers should consider students' developmental stages and prior knowledge, design tasks with appropriate scaffolding and sequencing, simplify complex concepts, offer clear instructions, and foster opportunities for exploration.

▪ ACKNOWLEDGMENT

This research was funded by the Universitas Pendidikan Indonesia through the Directorate of Research and Community Service in 2024.

▪ REFERENCES

Anderson, J. R. (1982). Acquisition of cognitive skill. In Psychological Review (Vol. 89, Issue 4, pp. 369–406). American Psychological Association. https://doi.org/10.1037/0033-295X.89.4.369

Angeli, C., & Valanides, N. (2009). Epistemological and methodological issues for the conceptualization, development, and assessment of ICT–TPCK: Advances in technological pedagogical content knowledge (TPCK). Computers & Education, 52(1), 154–168. https://doi.org/https://doi.org/10.1016/j.compedu.2008.07.006

Asheela, E., Ngcoza, K. M., & Sewry, J. (2020). The use of easily accessible resources during hands-on practical activities in rural under-resourced Namibian schools. In School science practical work in Africa (pp. 14–31). Routledge.

Basu, S., Biswas, G., Sengupta, P., Dickes, A., Kinnebrew, J. S., & Clark, D. (2016). Identifying middle school students' challenges in computational thinking-based science learning. Research and Practice in Technology Enhanced Learning, 11(1). https://doi.org/10.1186/s41039-016-0036-2

Bati, K., Yetişir, M. I., Çalişkan, I., Güneş, G., & Gül Saçan, E. (2018). Teaching the concept of time: A steam-based program on computational thinking in science education. Cogent Education, 5(1), 1507306. https://doi.org/10.1080/2331186X.2018.1507306

Beckers, K., Faßbender, S., Heisel, M., & Paci, F. (2013). Combining goal-oriented and problem-oriented requirements engineering methods bt - availability, reliability, and security in information systems and HCI (A. Cuzzocrea, C. Kittl, D. E. Simos, E. Weippl, & L. Xu (Eds.); pp. 178–194). Springer Berlin Heidelberg.

Brousseau, G., Balacheff, N., Cooper, M., Sutherland, R., & Warfield, V. (Eds.). (2002). Epistemological obstacles, problems, and didactical engineering BT - Theory of didactical situations in mathematics: didactique des mathématiques, 1970–1990 (pp. 79–117). Springer Netherlands. https://doi.org/10.1007/0-306-47211-2_6

Camacho, D., Panizo-LLedot, Á., Bello-Orgaz, G., Gonzalez-Pardo, A., & Cambria, E. (2020). The four dimensions of social network analysis: An overview of research methods, applications, and software tools. Information Fusion, 63(1), 88–120. https://doi.org/10.1016/j.inffus.2020.05.009

Chevallard, Y. (2007). Readjusting didactics to a changing epistemology. european educational Research Journal, 6(2), 131–134. https://doi.org/10.2304/eerj.2007.6.2.131

Chevallard, Y., Barquero, B., Bosch, M., Florensa, I., Gascón, J., Nicolás, P., & Ruiz-Munzón, N. (2022). Advances in the Anthropological Theory of the Didactic. Advances in the Anthropological Theory of the Didactic, 1–352. https://doi.org/10.1007/978-3-030-76791-4

Chevallard, Y., & Bosch, M. (2020). Didactic transposition in mathematics education BT - encyclopedia of mathematics education (S. Lerman (Ed.); pp. 214–218). Springer International Publishing. https://doi.org/10.1007/978-3-030-15789-0_48

Cholilah, M., Tatuwo, A. G. P., Komariah, & Rosdiana, S. P. (2023). *Pengembangan kurikulum merdeka dalam satuan pendidikan serta implementasi kurikulum merdeka pada pembelajaran abad 21*. Sanskara Pendidikan Dan Pengajaran, 1(02), 56–67. https://doi.org/10.58812/spp.v1i02.110

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2022). Introduction to Algorithm. In MIT press. (Vol. 2). https://doi.org/10.4324/9780429480898

Creswell, J. W., & Creswell, J. D. (2017). Research design: qualitative, quantitative, and mixed methods approaches. Sage publications.

Dagiene, V., & Dolgopolovas, V. (2022). Short tasks for scaffolding computational thinking by the global bebras challenge. Mathematics, 10(17). https://doi.org/10.3390/math10173194

Dagienė, V., & Sentance, S. (2016). It's computational thinking! bebras tasks in the curriculum bt - informatics in schools: improvement of informatics knowledge and perception (A. Brodnik & F. Tort (Eds.); pp. 28–39). Springer International Publishing.

Delal, H., & Oner, D. (2020). Developing middle school students' computational thinking skills using unplugged computing activities. Informatics in Education, 19(1), 1–13.

Dwiputra, D. F. K., Azzahra, W., & Heryanto, F. N. (2023). A Systematic Literature Review on Enhancing the Success of Independent Curriculum through Brain-Based Learning Innovation Implementation. Indonesian Journal on Learning and Advanced Education (IJOLAE), 5(3), 262–276. https://doi.org/10.23917/ijolae.v5i3.22318

Elo, S., & Kyngäs, H. (2008). The qualitative content analysis process. Journal of Advanced Nursing, 62(1), 107–115. https://doi.org/https://doi.org/10.1111/j.1365-2648.2007.04569.x

González, A., Gallego-Sánchez, I., Gavilán-Izquierdo, J. M., & Puertas, M. L. (2021). Characterizing levels of reasoning in graph theory. Eurasia Journal of Mathematics, Science and Technology Education, 17(8), 1–16. https://doi.org/10.29333/ejmste/11020

Hakiky, N., Nurjanah, S., & Fauziati, E. (2023). *Kurikulum merdeka dalam perspektif filsafat konstruktivisme.* Tsaqofah, 3(2), 194–202. https://doi.org/10.58578/tsaqofah.v3i2.887

Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. Computers & Education, 126, 296–310. https://doi.org/10.1016/j.compedu.2018.07.004

Hudha, M. N., Etikamurni, D. P., Rasuli, I. P. A., Safitri, N. S., Perwita, C. A., & Ayu, H. D. (2023). Problem based learning (PBL)-problem solving skills (PSS): systematic literature review. IJIS Edu: Indonesian Journal of Integrated Science Education, 5(2), 103-116.

Jiang, B., & Li, Z. (2021). Effect of Scratch on computational thinking skills of Chinese primary school students. Journal of Computers in Education, 8(4), 505–525. https://doi.org/10.1007/s40692-021-00190-z

Juliangkary, E., Suastra, I. W., & Atmaja, A. W. T. (2023). Kurikulum Merdeka: Filsafat Pendidikan dan Nilai-Nilai Ki Hajar Dewantara dalam Sorotan Kritis. Empiricism Journal, 4(2), 598–605. https://journal-center.litpam.com/index.php/empiricism/article/view/1665

Lamprou, A., & Repenning, A. (2018). Teaching how to teach computational thinking. Annual Conference on Innovation and Technology in Computer Science Education, ITiCSE, 69–74. https://doi.org/10.1145/3197091.3197120

Lehtimäki, T., Hamm, J., Mooney, A., Casey, K., Monahan, R., & Naughton, T. J. (2022). A computational thinking module for secondary students and pre-service teachers using Bebras-style tasks. Proceedings of the 2022 Conference on United Kingdom \& Ireland Computing Education Research. https://doi.org/10.1145/3555009.3555030

Leova, L., Cubanova, S., Kutilek, P., Volf, P., Hejda, J., Hybl, J., Stastny, P., Vagner, M., & Krivanek, V. (2022). Current state and design recommendations of exoskeletons of lower limbs in military applications BT - Modelling and simulation for autonomous systems (J. Mazal, A. Fagiolini, P. Vasik, M. Turi, A. Bruzzone, S. Pickl, V. Neumann, & P. Stodola (Eds.); pp. 452–463). Springer International Publishing.

Lockwood, J., & Mooney, A. (2018). Computational thinking in secondary education: where does it fit? A systematic literary review. International Journal of Computer Science Education in Schools, 2(1), 41–60. https://doi.org/10.21585/ijcses.v2i1.26

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

Nabilah, B., Zakir, S., Murtiyastuti, E., & Mubaraq, R. I. (2023). Analisis Penerapan Mata Pelajaran Informatika dalam Implementasi Kurikulum Merdeka Tingkat SMP. PIJAR: Jurnal Pendidikan Dan Pengajaran, 1(1 SE-Articles), 110–119. https://doi.org/10.58540/pijar.v1i1.97

Nagata, K., & Nishi, T. (2021). Modeling object arrangement patterns and picking arranged objects. Advanced Robotics, 35(16), 981–994. https://doi.org/10.1080/01691864.2021.1948446

Nazihah, Z., & Nurcahyo, D. (2019). Developing concept construction based abstract algebra textbook to improve students' understanding. Eduma : Mathematics Education Learning and Teaching, 8(2), 36. https://doi.org/10.24235/eduma.v8i2.4912

Pan, Y., Adams, E. L., Ketterlin-Geller, L. R., Larson, E. C., & Clark, C. (2024). Enhancing middle school students' computational thinking competency through game-based learning. Educational Technology Research and Development. https://doi.org/10.1007/s11423-024-10400-x

Pansell, A. (2023). Mathematical knowledge for teaching as a didactic praxeology. Frontiers in Education, 8. https://www.frontiersin.org/journals/education/articles/10.3389/feduc.2023.1165977

Pata'dungan, C. P., Pilemon, M., Rapang, N., & Mangguali, M. (2023). *Penggunaan asas-asas filosofis dalam pengembangan dan pelaksanaan kurikulum merdeka.* Jurnal Pendidikan Dan Keguruan, 1(6), 555–566.

Pérez, A. (2018). A framework for computational thinking dispositions in mathematics education. Journal for Research in Mathematics Education JRME, 49(4), 424–461. https://doi.org/10.5951/jresematheduc.49.4.0424

Polat, E., Hopcan, S., & Arslantaş, T. K. (2022). The association between flipped learning readiness, engagement, social anxiety, and achievement in online flipped classrooms: a structural equational modeling. Education and Information Technologies, 27(8), 11781–11806. https://doi.org/10.1007/s10639-022-11083-8

Putra, Z. H., Dahnilsyah, & Aljarrah, A. (2021). A Praxeological analysis of pre-service elementary teacher-designed mathematics comics. Journal on Mathematics Education, 12(3), 563–580. https://doi.org/10.22342/jme.12.3.14143.563-580

Reiser, B. J. (2018). Scaffolding complex learning: the mechanisms of structuring and problematizing student work. Scaffolding: A Special Issue of the Journal of the Learning Sciences, 13(3), 273–304. https://doi.org/10.4324/9780203764411-2

Rich, P. J., Egan, G., & Ellsworth, J. (2019). A Framework for decomposition in computational thinking. proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education, 416–421. https://doi.org/10.1145/3304221.3319793

Rønning, F. (2022). Mathematics and engineering : Interplay between praxeologies. 1, 1–4.

Sidik, G. S., Suryadi, D., & Turmudi, T. (2021). Learning obstacle on addition and subtraction of primary school students: analysis of algebraic thinking. Education Research International, 2021(1), 5935179. https://doi.org/10.1155/2021/5935179

Suryadi, D. (2019). *Landasan filosofis penelitian DDR.* Pusat Pengembangan DDR Indonesia.

Takeuchi, H., & Shinno, Y. (2020). Comparing the lower secondary textbooks of japan and england: a praxeological analysis of symmetry and transformations in geometry. International Journal of Science and Mathematics Education, 18(4), 791–810. https://doi.org/10.1007/s10763-019-09982-3

Tsai, M.-J., Liang, J.-C., Lee, S. W.-Y., & Hsu, C.-Y. (2021). Structural validation for the developmental model of computational thinking. Journal of Educational Computing Research, 60(1), 56–73. https://doi.org/10.1177/07356331211017794

Utami, N. S., Prabawanto, S., & Suryadi, D. (2024). How do Indonesian students learn function concepts? A praxeological analysis of textbook. Journal on Mathematics Education, 15(2), 451–472. https://doi.org/10.22342/jme.v15i2.pp451-472

van de Pol, J., Mercer, N., & Volman, M. (2019). Scaffolding student understanding in small-group work: students' uptake of teacher support in subsequent small-group interaction. Journal of the Learning Sciences, 28(2), 206–239. https://doi.org/10.1080/10508406.2018.1522258

Vermunt, J. D. (2006). Balancing support for student learning. In Handling complexity in learning environments: Theory and research (pp. 167–184).

Wijanto, M. C., Wisnubhadra, I., Natali, V., Wahyono, Mulyati, S., Wardhani, A., Sutardi, Pratiwi, H., Saputra, B., Astiani, K., & Sumiati. (2021). *Informatika untuk SMP Kelas VII.* Pusat kurikulum dan perbukuan badan penelitian dan pengembangan dan perbukuan kementerian pendidikan, kebudayaan, riset, dan teknologi.

Wijayanti, D., & Winslow, C. (2017). Mathematical practice in textbooks analysis: Praxeological reference models, the case of proportion. Journal of Research in Mathematics Education, 6(3), 307–330. https://doi.org/10.17583/redimat.2017.2078

Wisnubhadra, I., Wijanto, M. C., Natali, V., Wahyono, W., Mulyati, S., Wardhani, A., Sutardi, S., Pratiwi, H., Saputra, B., Astiani, K., & Sumiati, S. (2021). *Buku panduan guru informatika SMP Kelas VII.* Pusat Kurikulum dan Perbukuan Badan

Penelitian dan Pengembangan dan Perbukuan Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi.

Yin, R. K. (2016). Qualitative research from start to finish. In Guildford Press: Vol. 2nd Ed. Guilford publications.

Yunianta, T. N. H., Suryadi, D., Dasari, D., & Herman, T. (2023). Textbook praxeological-didactical analysis: Lessons learned from the Indonesian mathematics textbook. Journal on Mathematics Education, 14(3 SE-Articles), 503–524. https://doi.org/10.22342/jme.v14i3.pp503-524