MathEdu Journal of the Korean Society of Mathematical Education Series A

Research Article

The Mathematical Education 2024;63(2):233-254
https://doi.org/10.7468/mathedu.2024.63.2.233
pISSN 1225-1380, eISSN 2287-9633

Unveiling the synergistic nexus: AI-driven coding integration in mathematics education for enhanced computational thinking and problem-solving

Ipek Saralar-Aras*, Yasemin Cicek Schoenberg

National Education Expert, Ministry of National Education

ABSTRACT

This paper delves into the symbiotic integration of coding and mathematics education, aimed at cultivating computational thinking and enriching mathematical problem-solving proficiencies. We have identified a corpus of scholarly articles (n=38) disseminated within the preceding two decades, subsequently culling a portion thereof, ultimately engendering a contemplative analysis of the extant remnants. In a swiftly evolving society driven by the Fourth Industrial Revolution and the ascendancy of Artificial Intelligence (AI), understanding the synergy between these domains has become paramount. Mathematics education stands at the crossroads of this transformation, witnessing a profound influence of AI. This paper explores the evolving landscape of mathematical cognition propelled by AI, accentuating how AI empowers advanced analytical and problem-solving capabilities, particularly in the realm of big data-driven scenarios. Given this shifting paradigm, it becomes imperative to investigate and assess AI's impact on mathematics education, a pivotal endeavor in forging an education system aligned with the future. The symbiosis of AI and human cognition doesn't merely amplify AI-centric thinking but also fosters personalized cognitive processes by facilitating interaction with AI and encouraging critical contemplation of AI's algorithmic underpinnings. This necessitates a broader conception of educational tools, encompassing AI as a catalyst for mathematical cognition, transcending conventional linguistic and symbolic instruments.

Keywords Algorithmic thinking, Artificial intelligence, Computer programming, Innovative pedagogies in mathematics, Learning and teaching mathematics

Introduction

In an era defined by the rapid advancement of technology and the transformative wave of the Fourth Industrial Revolution, the intersection of academic disciplines has gained unprecedented significance as a means to equip individuals with the necessary skills and competencies to navigate the complexities of the modern world (Sáez-López et al., 2019; Weintrop et al., 2016). Among these disciplines, the fusion of coding

Received December 4, 2023; **Revised** February 14, 2024; **Accepted** April 12, 2024

***Corresponding author** Ipek Saralar-Aras
E-mail ipek.saralararas@gmail.com
2000 Mathematics Subject Classification 97B10, 97C80

and mathematics education has emerged as a particularly promising avenue for nurturing essential skills such as computational thinking and mathematical problem-solving (Francies et al., 2016; Francis & Davis, 2018). This paper endeavors to delve deep into the symbiotic integration of coding and mathematics education, with a specific focus on how this integration can not only foster the development of computational thinking skills but also significantly enhance proficiency in solving intricate mathematical problems.

As we traverse an increasingly intricate landscape shaped by the ascendancy of Artificial Intelligence (AI) and the Fourth Industrial Revolution, the convergence of coding and mathematics becomes not just relevant but imperative (Miller, 2019; Popat & Starkey, 2019). The Fourth Industrial Revolution, characterized by the seamless fusion of digital, biological, and physical technologies, along with the widespread proliferation of AI, has fundamentally redefined the skill sets required to not only thrive but also to remain relevant in this new paradigm (Psycharis & Kallia, 2017; Sun et al., 2021). In this context, computational thinking emerges as a pivotal skill set, characterized by elements such as algorithmic reasoning, abstraction, pattern recognition, and problem decomposition (Rich et al., 2020; Zhang et al., 2021). These elements collectively form the cognitive toolkit that individuals need to analyze complex problems and formulate innovative solutions - a skill set demanded by the rapidly evolving landscape (diSessa, 2018; Ye et al., 2023). Concurrently, mathematics retains its position as a cornerstone of education, providing the logical framework essential for understanding and harnessing the potential of these technological advancements.

This paper seeks to embark on an exploration of how the symbiotic merger of coding and mathematics education can harness the formidable power of AI to propel cognitive development. With AI's unparalleled ability to analyze voluminous datasets and solve intricate problems, its role as a catalyst for augmenting both coding and mathematical capabilities becomes distinctly apparent (Mohamadou et al., 2020; Moore & Chow, 2021). The impact of AI on mathematical cognition stretches far beyond mere automation; it ushers in new avenues for critical thinking and intellectual engagement, as individuals interact with AI algorithms and delve into the intricate underpinnings of these algorithms to unravel their underlying mechanisms, which are considered to be a part of mathematics education (Deeva et al., 2021; Lin et al., 2013).

The shifting educational landscape of our time necessitates a comprehensive reevaluation of pedagogical approaches (Adanır et al., 2023; Kado, 2022). This paper intends to illuminate the evolving interplay between AI, coding, and mathematics education, offering profound insights into how these seemingly distinct domains can synergistically shape the future of learning. By meticulously investigating the intersections of AI, computational thinking, and mathematical problem-solving, we aspire to make a valuable contribution to the ongoing discourse surrounding the design of an education system that harmonizes seamlessly with the demands of the future.

1. Literature Review

The integration of coding and mathematics education has attracted escalating attention over the past few decades, buoyed by the increasing recognition of their inherent compatibility and potential to nurture crucial skills in students (diSessa, 2018; Ye et al., 2023). A substantial and diverse body of scholarly work has emerged during this period, spanning the last two decades, that meticulously explores the potential symbiosis between these two domains (Perez, 2018; Weintrop et al., 2016). Through a methodical review and synthesis of these scholarly contributions, this literature review seeks to provide a comprehensive and nuanced understanding of the evolving landscape at the juncture of coding, mathematics, and education.

Firstly, multiple studies suggest that integrating computational thinking and programming into mathematics education can positively impact students' mathematical learning and achievement (Ardito et al., 2014; Chan et al., 2023; Ye et al., 2023). Ardito et al. (2014) explored the incorporation of a technology-rich learning environment in the context of STEM education, particularly focusing on robotics integration in a sixth-grade mathematics classroom. Through a collaborative effort between a middle school and a university technology program, students engaged in robot challenges that fostered collaborative problem-solving. Analysis of state-mandated mathematics exam scores revealed that students deeply involved in the robotics program achieved higher scores in algebra, measurement, and probability, reflecting their enhanced problem-solving abilities. This innovative approach highlights the potential of technology-rich STEM learning experiences to improve both mathematical understanding and collaborative skills.

Building on this notion, Ye et al. (2023) delved into the extensive research landscape surrounding the integration of computational thinking in K-12 mathematics education. Their systematic review emphasized the need for clear articulation of how computational thinking supports mathematics learning in CT-based mathematics activities. They identified the symbiotic relationship between computational thinking and mathematics through the application of mathematical concepts to construct computational artifacts, interpret computational outputs, and generate new mathematical knowledge. This interactive and cyclical process underscores the productive learning outcomes that emerge from the integration of computational thinking and mathematics instruction.

In a scoping review, Chan et al. (2023) furthered the understanding of learning tools for integrating computational thinking and mathematics. Their examination of empirical studies from 2015 to 2021 revealed the prevalence of CT-intensive Math-connected integration. They identified five major types of CT tools, including digital tangibles, apps and games, programming languages, formative or summative assessments, and other technological tools. The review highlighted the assessment of critical CT competencies such as algorithms, abstraction, testing and debugging, loops, and sequences. Geometry and Measurement emerged as the most frequently assessed mathematics topics, emphasizing the potential for enriching mathematical understanding through targeted integration of computational thinking tools.

Similarly, Darmawansah et al. (2023) explored the role of project-based learning in the context of Robotics-based STEM (R-STEM) education. Their systematic review of 39 articles revealed that project-based learning was the most frequently employed learning strategy in robotics-related STEM research. The study emphasized the positive impact of project-based learning on fostering students' competence in applying interdisciplinary knowledge to solve problems within the realm of STEM education. The findings underscored the prevalence of project-based learning as a valuable approach in robotics-related STEM studies, contributing to learners' understanding and transferable skills (Darmawansah et al., 2023). Collectively, these studies underscore the manifold benefits of integrating computational thinking and programming into mathematics education.

Another prevailing and recurrent theme within this extensive literature is the pivotal role that computational thinking plays in shaping students' problem-solving abilities. Computational thinking, deeply rooted in coding practices, revolves around the systematic breakdown of complex challenges into more manageable steps, followed by the design and application of algorithmic approaches to tackle these constituent components. Researchers and educators alike (Gadanidis et al., 2018; Zhang et al., 2023) have consistently emphasized the transformative potential of teaching coding principles in enhancing students' capacity to dissect intricate mathematical problems. The explicit connection between coding and mathematical thinking has been shown to promote structured, systematic, and methodical approaches to problem-solving, thereby accentuating the transferable nature of these skills across disciplines.

Yet, the advent of AI has injected an unprecedented transformative dimension into this symbiotic relationship between coding and mathematics. AI's exceptional capacity to process, analyze, and derive insights from massive and diverse datasets has profound implications for mathematical cognition (Saralar-Aras et al., 2023). Particularly in the realm of big data-driven scenarios, AI emerges as a powerful augmentation tool for human intellectual capabilities. Researchers (Mohamadou et al., 2020; Moore & Chow, 2021) have elaborated on how AI's capacity to identify correlations and uncover hidden relationships resonates with the principles of mathematical modeling and analysis. This intersection underscores the potential of AI to not only expedite computational processes but also to catalyze the development of advanced mathematical thinking, enabling individuals to engage with complex real-world problems that are characterized by an abundance of data.

Moreover, the role of AI in personalized education and the facilitation of individualized learning pathways has gained significant traction within the literature (Saralar-Aras, 2021). By harnessing AI-driven platforms, educators can provide tailored and adaptive learning experiences that are optimized for each learner's cognitive profile and pace of understanding (Deeva et al., 2021; Lin et al., 2013). This underscores the evolving role of AI as a true partner in the educational journey, not only assisting in the transmission of knowledge but also stimulating critical reflection and deepening conceptual comprehension.

In summary, the seamless integration of coding and mathematics education, fortified by the transformative potential of AI, signifies a profound paradigm shift in pedagogical approaches. This expansive literature review serves as a solid foundation for the subsequent contemplative analysis of the extant remnants within the domain. More importantly, it underscores the imperative of cultivating computational thinking and enhancing mathematical problem-solving proficiencies as essential skills to effectively navigate the complexities of the Fourth Industrial Revolution and the AI-driven landscape of our future.

2. Significance of the Study

The integration of coding and mathematics education has become a focal point in educational research, driven by the recognition of the synergies between these two domains and their potential to cultivate essential skills in students (Chan et al., 2023; Ye et al., 2023). Over the past few decades, a substantial body of literature has emerged, exploring various aspects of this integration, from theoretical frameworks to practical instructional strategies (Darmawansah et al., 2023). This study conducted a comprehensive literature review, drawing from reputable electronic databases (ACM Digital Library, EBSCO [Elton B. Stephens Company], ERIC [Education Resources Information Center], Google Scholar, and IEEE Xplore) and utilizing stringent inclusion criteria to identify relevant research studies, theoretical frameworks, and instructional approaches. By synthesizing these scholarly contributions, the review aims to offer a nuanced understanding of the advantages, challenges, and strategies associated with the integration of coding and mathematics education.

One recurrent theme highlighted in the literature is the significant role of computational thinking in enhancing students' problem-solving abilities within the context of mathematics education (Bråting & Kilhamn, 2021; Grover et al., 2022). Computational thinking, deeply rooted in coding practices, involves breaking down complex problems into manageable steps and applying algorithmic approaches to solve them systematically. Studies have consistently shown that teaching coding principles not only fosters computational thinking but also promotes structured and methodical problem-solving strategies in mathematics (Acar & Övez, 2022; Adanır et al., 2023; Kampylis et al., 2023; Zhang et al., 2023). This explicit connection between coding and mathematical thinking underscores the transferability of these skills across disciplines, emphasizing their relevance in preparing students for the challenges of the 21st-century workforce. Through its meticulous review and synthesis of existing research, this study contributes to the literature by offering valuable insights into the evolving landscape of coding, mathematics, and education, thereby informing educators and researchers about effective approaches to integrate these domains in educational practice.

3. Theoretical Framework

This paper adopts a theoretical framework grounded in cognitive psychology, educational technology, and pedagogical theory to elucidate the symbiotic integration of coding and mathematics education. Drawing on principles of constructivism (Troelstra, & Van Dalen, 2014), cognitive load theory (Phan et al., 2017), and situated cognition (Brown et al., 1989), the framework emphasizes the active construction of knowledge through meaningful engagement with coding activities in mathematical contexts.

Constructivism posits that learners actively construct knowledge by assimilating new information into existing cognitive structures and accommodating these structures through interaction with the environment (Steffe & Gale, 1995). In the context of coding and mathematics education, learners engage in problem-solving tasks, algorithmic thinking, and computational modeling, which facilitate the construction of mathematical concepts and problem-solving strategies (Troelstra, & Van Dalen, 2014).

Cognitive load theory provides insights into how cognitive resources are allocated during learning tasks, emphasizing the importance of managing cognitive load to optimize learning outcomes (Sweller, 2011). By scaffolding coding activities and providing instructional support, educators can mitigate extraneous cognitive load, allowing learners to focus on the development of computational thinking skills and mathematical problem-solving strategies (Phan et al., 2017).

Situated cognition theory posits that learning is situated within authentic contexts and social interactions, emphasizing the importance of contextualized learning experiences (Wilson & Myer, 2000). In the context of coding and mathematics education, authentic tasks such as coding projects, real-world problem-solving scenarios, and collaborative programming activities provide learners with opportunities to apply mathematical concepts and computational thinking skills in meaningful contexts (Brown et al., 1989).

By integrating these theoretical perspectives, this paper seeks to elucidate the cognitive processes underlying the integration of coding and mathematics education and to inform the design of effective instructional approaches and learning environments. We believe that through a deeper understanding of the theoretical underpinnings of this integration, educators can better support students' development of computational thinking skills and mathematical problem-solving abilities in the era of Artificial Intelligence and the Fourth Industrial Revolution.

4. Originality of the Study

This study contributes to the literature on the integration of coding and mathematics education by offering a distinct perspective and original insights. While previous literature reviews such as Chan et al. (2023) and Ye et al. (2023) have explored various aspects of this integration, our research provides a comprehensive analysis that goes beyond a mere juxtaposition of coding and mathematics. One notable aspect of our study is the elucidation of the distinction between coding and computational thinking (Bråting & Kilhamn, 2021; Israel & Lash, 2020). While coding involves the implementation of programming languages to create software or applications, computational thinking encompasses a broader set of problem-solving skills, including algorithmic thinking, abstraction, and decomposition. By clarifying this distinction, our research offers a more nuanced understanding of how computational thinking can be integrated into mathematics education beyond coding activities alone.

Furthermore, our study goes beyond the surface level of mathematics education to delve into the realm of mathematical problem-solving, which have been reported to be in need by various researchers (Çam & Kıyıcı, 2022; Psycharis & Kallia, 2017). In our context, while mathematics education encompasses the teaching and learning of mathematical concepts and procedures, mathematical problem-solving involves the application of these concepts in real-world contexts to solve complex problems. By focusing on the intersection of coding and mathematical problem-solving, our research highlights the potential of coding activities to enhance students' problem-solving skills within the domain of mathematics. This emphasis on problem-solving not only enriches the educational experience but also prepares students with the critical thinking and analytical skills necessary for success in today's technology-driven world (Çiftci & Bildiren, 2020; Psycharis & Kallia, 2017). In essence, our study offers a fresh perspective on the integration of coding and mathematics education, emphasizing the importance of computational thinking and problem-solving skills in preparing students for the challenges of the future.

Methodology

This study undertook a comprehensive literature review to investigate the integration of coding and mathematics education. The review aimed to identify relevant research studies, theoretical frameworks, best practices, and instructional approaches contributing to the comprehension of the advantages, challenges, and strategies associated with the amalgamation of coding and mathematics.

1. Search Process

To ensure the inclusivity of recent and up-to-date studies, an extensive search process was conducted using reputable electronic databases, including ACM Digital Library, EBSCO, ERIC, Google Scholar, and IEEE Xplore. The search utilized keywords and phrases such as "coding and mathematics integration," "computational thinking in mathematics education," "programming in mathematics instruction," and variations such as "math," "maths," "mathematic," and "mathematics." The search was limited to studies published in English from 2014 to the present.

2. Rationale for Search Boundary

The decision to set 2014 as the boundary for the search was based on several factors. Firstly, this timeframe allowed us to capture recent developments and trends in the integration of coding and mathematics education. Secondly, it ensured the relevance and currency of the findings, considering significant advancements in educational technology and pedagogy since that time. Additionally, 2014 marked a significant milestone in the recognition of CT as a core skill in education, with various initiatives and research efforts emerging in this area.

3. Inclusion and Exclusion Criteria

The selection of articles for this literature review was guided by stringent inclusion criteria designed to ensure the relevance, quality, and rigor of the research included in the analysis. Priority was accorded to papers deemed well-qualified, particularly those that have undergone peer review. To ascertain the credibility and scholarly merit of the selected papers, several factors were taken into consideration, including the reputation of the publishing journals or conferences, the expertise of the authors, and the rigor of the research methods employed. Articles such as book reviews were intentionally excluded from consideration, as they did not align with the primary focus of the review. Specific inclusion and exclusion criteria were established to guide the selection process:

- Participants: The literature review focused on studies involving K12 students, ensuring that the findings were relevant to the target audience of mathematics educators and practitioners working with students in primary and secondary education settings. Those articles having participants such as pre-service teachers and teachers were excluded.
- Subject Matter: Articles were selected based on their focus on teaching and learning a particular mathematics subject. This criterion ensured that the research under consideration directly addressed the integration of computational thinking, mathematics education, and programming within the context of specific mathematical concepts or topics. Those articles including other subjects such as science and engineering in

focus were excluded.

• Language: Articles published in English were included in the review to ensure accessibility to a broad audience of researchers, educators, and practitioners in the field of mathematics education. Those articles in other languages were excluded. This criterion facilitated the synthesis and dissemination of findings to a global audience.

• Format: Only full papers were considered for inclusion in the literature review. Conference abstracts and book reviews were excluded from the research. This criterion ensured that the selected articles provided comprehensive coverage of the research conducted, including detailed descriptions of the theoretical frameworks, methodologies, findings, and implications for practice.

By adhering to these rigorous inclusion criteria, the literature review aimed to select high-quality research articles that provided valuable insights into the integration of computational thinking, mathematics education, and programming. This approach ensured that the findings synthesized in the review were robust, credible, and relevant to the needs and interests of educators and researchers in the field.

4. Screening and Evaluation Process

A systematic screening process was employed to review and assess articles for relevance and quality. Initially, a broad range of articles (n=1,026) was identified based on the relevance of their titles and abstracts. Priority was given to articles (n=38) that specifically addressed the integration of coding and mathematics education, examined the impact on students' computational thinking and mathematical problem-solving skills, and provided practical strategies or instructional approaches. Preference was given to articles presenting empirical research, case studies, and conceptual frameworks.

5. Data Analysis

The data analysis procedure was conducted meticulously, following a structured approach aligned with our theoretical and conceptual framework. We systematically examined the selected papers to identify prevalent themes, discern key findings, and extract insightful perspectives. Drawing from the theoretical underpinnings of cognitive psychology, educational technology, and pedagogical theory, we developed an analytical framework to categorize the collected data into four distinct domains, each serving as a lens through which to interpret the integration of coding and mathematics education: impact of CT on learning and achievement (1), level of integration of CT practices (2), use of computational tools (3), and integration of unplugged programming in non-computer science subjects (4).

The first domain focused on assessing the impact of computational thinking (CT) on learning outcomes and academic achievement within the context of mathematics education. We examined how the infusion of CT principles influenced students' problem-solving abilities, mathematical reasoning skills, and overall academic performance.

The second domain explored the level of integration of CT practices into mathematics education. This involved scrutinizing the extent to which coding activities, algorithms, and computational problem-solving methodologies were incorporated into the pedagogical practices of mathematics educators.

The third domain centered on the utilization of computational tools in mathematics instruction. We analyzed the prevalence and effectiveness of various digital tools, software applications, and programming languages employed to facilitate the teaching and learning of mathematical concepts.

Lastly, the fourth domain examined the integration of unplugged programming activities in non-computer science subjects. Here, we investigated how hands-on, offline coding exercises and computational thinking tasks were integrated into diverse educational contexts beyond traditional computer science classrooms.

Each domain underwent a rigorous analysis process, wherein we extracted relevant information, identified recurring patterns, and elucidated emerging trends across the reviewed studies. By employing this systematic approach, we were able to derive comprehensive insights into the multifaceted relationship between coding, computational thinking, and mathematics education, thereby enriching our understanding of this symbiotic integration.

6. Key Focus Areas

The literature review highlighted several key focus areas, including (1) Theoretical foundations supporting the integration of coding and mathematics education, elucidating the relationships between computational thinking, problem-solving, and mathematics; (2) Benefits associated with coding integration, including increased student engagement, improved critical thinking skills, and enhanced conceptual understanding of mathematical concepts; (3) Challenges associated with implementing coding activities in mathematics instruction, such as the need for teacher professional development, access to appropriate technology, and curriculum alignment; and, (4) Various instructional approaches and strategies for integrating coding into mathematics education, including project-based learning, coding platforms and tools, coding challenges and competitions, and interdisciplinary curriculum design. In summary, this literature review provides a comprehensive overview of the current state of knowledge regarding the integration of coding and mathematics education. The findings and insights derived from this review serve as a foundational resource for further exploration and understanding of effective practices in leveraging coding to enhance computational thinking and mathematical problem-solving skills among students.

Results

The reviewed literature covers a wide range of research studies that investigate the integration of computational thinking, mathematics education, and programming in various educational settings. These studies offer valuable insights into the impacts of incorporating computational thinking and programming into mathematics education. The following key themes and findings emerged from the reviewed research:

1. Impact on Mathematical Learning and Achievement

The mathematical learning and achievement domain primarily focus on the impact of computational thinking (CT) on academic performance and learning outcomes in the context of mathematics education. This domain delves into how the integration of CT practices influences students' proficiency in mathematical concepts, problem-solving abilities, and overall academic achievement in mathematics. It assesses the effectiveness of incorporating CT principles into mathematics instruction, the level of integration of CT practices within the curriculum, the utilization of computational tools to enhance mathematical learning, and the integration of unplugged programming activities in non-computer science subjects. Essentially, this domain examines how CT contributes to students' success in academic mathematics.

The exploration of the impact of computational thinking on mathematical learning and achievement reveals a multifaceted landscape shaped by various studies. As we delve into the distinct facets of this intersection, it becomes evident that the integration of computational thinking practices into mathematics education is a dynamic and nuanced process. This synthesis of research findings unfolds across three key dimensions: the impact of computational thinking on learning and achievement, the level of integration of computational thinking practices, and the use of computational tools. Additionally, emerging trends shed light on the integration of "unplugged" approaches in non-computing disciplines, providing a holistic understanding of the evolving paradigms that shape contemporary educational practices. Each dimension brings forth valuable insights that contribute to the ongoing discourse on fostering computational thinking skills and enriching mathematical education.

(1) Level of Integration of Computational Thinking Practices

The intricate relationship between the level of integration of computational thinking practices into mathematics education and the depth of students' comprehension of the subjects is a critical consideration (Israel & Lash, 2020). Achieving an optimal balance is pivotal; if not established appropriately, there is a risk that students might not fully engage with essential mathematical concepts due to the potential encumbrance of cumbersome computer programming procedures (Bråting & Kilhamn, 2021). This delicate equilibrium resonates with the findings of various studies, such as those by Israel and Lash (2020), and Grover et al. (2022), each shedding light on the multifaceted nature of integration.

Israel and Lash (2020) delved into the integration of computer science and computational thinking into elementary mathematics instruction. They examined the relative amount of instructional time allocated to each discipline and uncovered varying degrees of integration, ranging from no integration to full integration. The research emphasized the need for a coherent transition from less integrated to more integrated activities, anchored by an initial focus on discipline-specific conceptual understanding. These findings underscored the intricate interplay between computational thinking and mathematics instruction, necessitating a calibrated approach to ensure that the balance between the two is conducive to enhanced learning outcomes.

Grover et al. (2022) extended this discourse to the early childhood context, where computational thinking intersects with early mathematics and science learning goals. Through collaborative efforts involving preschool teachers, families, and children, they identified specific CT skills like problem decomposition, algorithmic thinking, abstraction, testing, and debugging as foundational for integration. The study illuminated the potential of integrating CT skills with math and science concepts in early learning experiences, suggesting that judicious alignment of these elements can create powerful educational synergies. However, the delicate challenge of maintaining an appropriate balance to avoid overwhelming young learners remained a focal point.

In essence, the level of integration of computational thinking practices into mathematics education is a dynamic factor that can significantly impact students' grasp of essential mathematical concepts. The experiences presented by Israel and Lash (2020), and Grover et al. (2022) collectively reinforce the notion that a well-calibrated integration approach, mindful of the balance between computational thinking and mathematics, is indispensable for fostering deeper understanding while mitigating the risk of potential disengagement due to procedural complexities.

(2) Use of Computational Tools

Lastly, the utilization of coding activities, robotics, and computational tools emerges as a compelling avenue for enhancing students' grasp of mathematical concepts and honing their prowess in problem-solving (Acar & Övez, 2022; Adanır et al., 2023; Kampylis et al., 2023; Zhang et al., 2023). Recent endeavors in education have witnessed a deliberate infusion of Computational Thinking (CT) principles into primary and secondary curricula across numerous nations. This trend is corroborated by a meticulous review of both academic and grey literature, which underscores the prevalence of a core set of pivotal CT skills within primary and lower secondary education (Kampylis et al., 2023). The methodologies for CT integration have been delineated in diverse forms, spanning from embedding CT throughout the curriculum as a transversal skill set to integrating it as a standalone subject or within specific subjects such as Mathematics and Technology (Kampylis et al., 2023). These diverse approaches reflect the concerted effort to empower students with CT skills in multifaceted ways, which resonate with various educational contexts and goals.

Moreover, the notion of CT transcends traditional boundaries and extends its influence into interdisciplinary domains. While studies have predominantly emphasized the integration of CT with science, technology, engineering, and mathematics (STEM), recent exploration has embraced the STEAM perspective, which encompasses science, technology, engineering, the arts, and mathematics (Zhang et al., 2023). This expansion heralds the significance of art in early CT learning and signifies a broader avenue for nurturing CT skills through cross-disciplinary engagement. As the exploration advances, it unveils the symbiotic relationship between CT and STEM, underscoring how CT serves as a catalyst that invigorates STEM learning for young learners (Zhang et al., 2023). This underscores the interconnectedness of subjects, where CT acts as a bridge, rendering intricate STEM concepts more accessible and engaging.

Furthermore, empirical research attests to the efficacy of utilizing coding activities, robotics, and computational tools to augment student learning outcomes. A notable study by Acar and Övez (2022) delved into the domain of block-based game development activities as a means to enhance geometry achievement and CT skills among seventh-grade students. This intervention-driven research employed a goal-based scenario approach, wherein a purposeful sampling of participants engaged in hands-on activities. The results of the study unveiled tangible enhancements in students' computational thinking skills and geometry achievement, illustrating the potency of such interactive approaches in fostering multifaceted cognitive growth (Acar & Övez, 2022).

Building on this notion, in the study conducted by Adanır et al. (2023), their comprehensive analysis unveiled that the predominant pedagogical methods employed in educational contexts primarily encompassed a diverse range of instructional strategies. These encompassed the utilization of robotics, block-based programming, STEM (Science, Technology, Engineering, and Mathematics) education, unplugged activities, and various forms of gaming-based learning activities. This rich array of pedagogical approaches highlighted the multifaceted nature of modern education and the dynamic ways in which educators engage with technology and interactive methodologies to enhance learning outcomes. With this wealth of insights in mind, it is conceivable that these findings could have significant implications for the integration of innovative teaching practices in various educational settings. Given the relevance and importance of this source, it is worth considering its incorporation into our current research or educational initiatives.

(3) Integration of Unplugged Programming in Non-CS Subjects

Lastly, Emerging trends suggest a growing interest in the utilization of "unplugged" approaches as a means to facilitate the integration of computational thinking (CT) into non-computing disciplines. These unplugged approaches offer several advantages, particularly for educators who lack a background in computer science or programming. By sidestepping the resource-intensive demands associated with traditional coding instruction, such as extensive time commitments and the need for professional development, as well as the challenges related to hardware management, unplugged methods present a more accessible entry point. As highlighted in a study by Huang and Looi (2021), this approach holds promise for supporting mathematics education by providing educators with tangible strategies for grounding abstract mathematical concepts, aligning with the foundational "Abstract" stage of the Pictorial-Concrete-Abstract framework.

In sum, the integration of coding activities, robotics, and computational tools into educational contexts offers a transformative paradigm for bolstering students' comprehension of mathematical concepts and their prowess in problem-solving. This approach mirrors the ever-evolving educational landscape's commitment to nurturing computational thinking skills, permeating both traditional and interdisciplinary domains. Through diverse integration methods and innovative pedagogical strategies, educators pave the way for holistic and enriched learning experiences that empower students with crucial skills for their academic and future pursuits.

2. Cognitive and Creative Development

As we delve into the realm of cognitive and creative development through computational thinking (CT) skills, the synthesis of research findings unfolds a rich tapestry of insights into how programming activities contribute to the intellectual growth of learners. This dimension is characterized by two key foci: the development of CT skills and the fostering of 21st-century skills. The former accentuates the pivotal role of programming in honing algorithmic thinking, pattern recognition, and logical reasoning, while the latter explores the broader impact of CT on creativity, critical thinking, and innovative problem-solving abilities. Through a comprehensive exploration of studies spanning various educational levels and contexts, this section illuminates the intricate connection between programming activities and the cognitive and creative development of students, shedding light on the transformative potential of computational thinking in shaping well-rounded, adept learners.

The cognitive and creative development domain extends beyond academic achievement to focus on broader cognitive skills and 21st-century competencies fostered through the integration of coding and mathematics education. This domain explores the development of CT skills, such as algorithmic thinking, logical reasoning, and problem-solving strategies, which are not only pertinent to mathematical tasks but also applicable to real-world challenges. Additionally, it emphasizes the cultivation of creativity, critical thinking, collaboration, communication, and adaptability—skills that are indispensable in the rapidly evolving landscape of the 21st century. Unlike the achievement domain, which primarily concerns academic outcomes in mathematics, the cognitive and creative development domain encompasses a broader spectrum of life skills and competencies essential for holistic student development and success in diverse personal, academic, and professional contexts.

(1) Development of CT Skills

Programming activities contribute to the development of essential computational thinking skills, encompassing algorithmic thinking, pattern recognition, and logical reasoning (Kado, 2022; Luo et al., 2022). The contemporary educational landscape recognizes the transformative potential of moder information and communication technology (ICT) tools in imparting mathematics and science education, particularly in regions witnessing rapid digital technology integration. A case in point is the utilization of Python programming, a dynamic tool capable of visually representing abstract mathematical concepts, which offers educators novel means to elucidate intricate calculus principles that were previously challenging to convey through traditional pedagogical methods (Kado, 2022). One illuminating study delves into the domain of Python-based coding's efficacy, specifically focusing on grade eleven students' understanding of calculus concepts. Employing a pre-test and post-test control group quasi-experimental design, the study showcased that the incorporation of Python-based coding activities led to a significant difference in favor of the experimental group, indicating enhanced comprehension of fundamental calculus principles (Kado, 2022).

Furthermore, the synergy between programming and mathematical learning is underscored by a pioneering initiative targeting younger learners. A math-infused computer science course designed for third to fifth-graders aimed to instill programming skills while reinforcing mathematical and computational concepts (Luo et al., 2022). Findings from this study underline the positive impact of such interventions on knowledge acquisition, showcasing noticeable improvements in students' grasp of mathematical and computational domains (Luo et al., 2022). This empirical exploration extends beyond academic outcomes, as it delves into students' perceptions and motivation. The study revealed that students found computer programming to be enjoyable, comprehensible, and valuable, thereby highlighting the potential of programming activities not only to foster cognitive growth but also to engender a positive attitude toward computational learning (Luo et al., 2022).

In sum, the burgeoning research on programming activities' influence on computational thinking underscores their role in nurturing critical skills that span algorithmic thinking, pattern recognition, and logical reasoning. The integration of dynamic programming tools, as evidenced by the Python-based coding initiatives, not only elucidates complex mathematical concepts but also enhances students' understanding and engagement. These efforts extend from higher secondary levels to early education, underlining the pervasive impact of programming activities on computational thinking skill development and mathematical comprehension.

(2) Fostering 21st-Century Skills

Computational thinking fosters students' creativity, critical thinking, and innovative problem-solving abilities (Bers et al., 2019; Mari et al., 2022). Addressing the dynamic interplay between spatial knowledge, robotics, and computer programming, a study delved into the potential of programmable floor robots, such as BeeBot®, to cultivate spatial knowledge in mathematics classrooms. Employing an observation grid that considered these distinct dimensions, the researchers analyzed mathematics lessons involving BeeBot® robots. The study's insights revealed an evolution across each dimension over the course of the lessons, showcasing how the integration of robotics and programming fosters the development of spatial understanding among young students (Mari et al., 2022).

In the realm of early childhood education, the fusion of coding, computational thinking, and robotics into formal curriculums stands as a compelling challenge. A study embracing the philosophy of "coding as a playground" within the framework of Positive Technological Development (PTD) harnessed the KIBO robotics kit to introduce coding and computational thinking to preschool children aged 3-5 years old. The results demonstrated the viability of teaching these foundational skills at an early age, ushering children as young as three into the world of computational literacy (Bers et al., 2019). Moreover, the study illuminated how such pedagogical interventions bolstered communication, collaboration, and creativity within the classroom, fostering a holistic learning environment. Notably, the educators exhibited enhanced autonomy and confidence in integrating coding and computational thinking across formal curricular activities, establishing connections with art, music, and social studies, thus amplifying the interdisciplinary impact of these skills (Bers et al., 2019). Finally, research showed evidence of the positive impact of block-based programming and robotics approaches on students' spatial reasoning ability (Dickson et al., 2022; Francis et al., 2016; Mari et al., 2022).

In conclusion, the integration of computational thinking into educational paradigms yields substantial dividends by fostering creativity, critical thinking, and problem-solving skills. Through the exploration of spatial knowledge integration with programmable floor robots and the incorporation of coding and computational thinking in early childhood education, these studies underscore the transformative potential of such approaches in cultivating cognitive prowess and holistic development among students.

3. Motivation, Interest, and Engagement

Numerous studies underscore the noteworthy benefits of integrating programming and computational activities with mathematics education. These findings consistently point to heightened levels of student motivation and engagement when these elements are combined (Darmawansah et al., 2023; Sümen, 2022).

Furthermore, the incorporation of programming environments and robotics into educational settings has been shown to not only pique students' interest in coding and mathematics but also cultivate positive attitudes towards STEM subjects and careers. This holistic approach contributes to bolstering students' self-efficacy in these fields (Daher, 2022; Hudson et al., 2020; Jiang et al., 2022).

These collective insights underscore the transformative potential of merging mathematics education with programming and robotics, heralding a new era of enhanced engagement, motivation, and career readiness in STEM disciplines.

4. Teacher Professional Development and Pedagogical Practices

Effective integration of computational thinking into education necessitates well-trained teachers who can skillfully design an appropriate level of integration within mathematics activities and effectively implement these strategies (Huang & Looi, 2021; Lavigne et al., 2020). To achieve this, it is imperative to establish professional development programs aimed at equipping educators with both computational thinking skills and innovative pedagogical approaches (Hickmott & Prieto-Rodriguez, 2018; Wang et al., 2016). These programs play a pivotal role in ensuring the successful implementation of computational thinking in the classroom, enabling teachers to confidently navigate this evolving educational landscape.

(1) Collaborative Learning

In the study conducted by Ardito et al. (2014), the focus was on investigating the impact of a robotics curriculum on students' mathematics achievement and their ability to collaborate effectively in problem-solving scenarios. The researchers unearthed compelling evidence that students who actively participated in collaborative problem-solving activities within the context of a robotics curriculum exhibited notably improved scores in key mathematical content areas. Specifically, these students demonstrated significant enhancements in their grasp of probability, algebra, and measurement concepts.

This study's findings underscore the potent synergy between robotics education and mathematics, highlighting how the incorporation of robotics can foster not only mathematical proficiency but also teamwork and problem-solving skills. Additionally, the importance of knowledge sharing in the context of education is further emphasized, as exemplified by the work of Forsström and Kaufmann (2018). Such knowledge sharing can serve as a catalyst for innovative educational approaches and the dissemination of best practices, ultimately benefiting both educators and students alike.

(2) Using Games as an Effective Pedagogical Strategy: The Realm of Game-Based Learning

The comprehensive review conducted in this study shed light on the adoption of game-based learning as a pedagogical approach within the realm of computer programming integrated with mathematics lessons, as evidenced by the research of Liu et al. (2014) and Ke (2014), among others. These scholars have delved into the exciting intersection of gaming and education.

Liu et al. (2014), for instance, ingeniously wove the captivating concept of space into their educational game, serving as an overarching theme that seamlessly bridged the realms of computer programming, physics, and mathematics. This innovative approach not only engaged students but also demonstrated the potential for interdisciplinary connections within game-based learning environments.

On the other hand, Ke (2014) opted for a design-based learning strategy when immersing students in mathematical concepts through games. This approach underscores the importance of design thinking and creative problem-solving in the learning process, offering students a dynamic platform to explore mathematical principles in a practical and engaging manner.

Furthermore, it is noteworthy that within the domain of game-based learning, certain game elements compel players to navigate conflicts embedded within the game's thematic structure. This not only enhances their cognitive skills but also encourages the development of crucial abilities such as memory retention, strategic thinking, and problem-solving, as elucidated by Amory et al. (1999). This multifaceted nature of game-based learning underscores its potential as a versatile pedagogical tool that can foster both content acquisition and the cultivation of essential cognitive and problem-solving skills.

(3) Project-Based Learning

The integration of project-based learning (PBL) and coding in educational settings has gained considerable attention as educators seek effective strategies to foster students' computer programming abilities and problem-solving skills. Wang et al. (2016) conducted a study comparing the effects of project-based computer programming activities on mathematics-gifted students and average students. The research proposed an integrated Scratch and project-based learning approach to infuse problem-solving scenarios into programming tasks. The results indicated that both groups showed significant progress, with mathematics-gifted students outperforming their peers in problem-solving performance, learning attitude, and motivation. This study highlighted the potential of PBL in enhancing the learning experiences of students across different proficiency levels.

Bråting et al. (2020) delved into the integration of programming in Swedish school mathematics, particularly in relation to algebra learning. The study, based on Chevallard's framework of the transposition of knowledge, investigated the activities and systems of representations introduced with the implementation of programming. Tentative results indicated syntactic and semantic differences between programming and algebra, posing potential challenges for students. The study shed light on the complexities of integrating coding into mathematics education and the need for careful consideration of the alignment between programming activities and mathematical concepts (Bråting et al., 2020).

Additionally, Calder (2018) focused on the use of Scratch, a graphical programming environment, to facilitate mathematical thinking among 10-year-old students. The research project revealed that Scratch provided opportunities for creative problem-solving and influenced the learning process. The findings suggested that teachers should not only use Scratch for developing coding skills but also recognize its potential to enhance thinking in other related areas, including mathematics (Calder, 2018).

In conlusion, the literature indicates that project-based learning, when integrated with coding activities, holds promise for enhancing students' problem-solving skills, learning attitudes, and motivation across various educational contexts. However, challenges such as syntactic and semantic differences between programming and mathematical concepts underscore the importance of thoughtful implementation and alignment in educational practices. As educators navigate the landscape of integrating coding and project-based learning, careful consideration of pedagogical strategies and their impact on students' cognitive development remains crucial.

In sum, the effective integration of computational thinking into education hinges on well-trained teachers equipped with both computational thinking skills and innovative pedagogical approaches. Professional development programs play a pivotal role in ensuring successful implementation in the classroom, enabling educators to confidently navigate this evolving educational landscape. Whether through collaborative learning, game-based strategies, or project-based learning, the literature underscores the potential of these approaches to enhance students' problem-solving skills, learning attitudes, and motivation. However, the challenges highlighted, such as syntactic and semantic differences in coding and mathematical concepts, emphasize the need for thoughtful implementation and alignment in educational practices as educators explore these innovative teaching methodologies.

5. Challenges and Future Directions

The integration of computational thinking and mathematics education holds immense promise, yet it is not without its share of challenges that warrant careful consideration (Bulut & Ferri, 2023; Popat & Starkey, 2019). These challenges encompass aspects such as aligning curricula, devising appropriate assessment methods, and ensuring equitable access to resources for all students.

In charting the course for future research in this domain, it becomes imperative to explore strategies that facilitate the widespread implementation of computational thinking initiatives while concurrently addressing issues of accessibility (Lavigne et al., 2020; Rich et al., 2020). Moreover, it is essential to expand the scope of curriculum studies beyond the commonly assessed areas, particularly given the findings of Chan et al. (2020) that suggest a need to investigate other mathematical content areas.

Delving deeper into the impact of programming activities on mathematical understanding, Laurent et al. (2022) uncovered negative effects on students' comprehension of concepts like Euclidean division, additive decomposition, and fractions. This underscores the need for a nuanced understanding of the potential drawbacks of integration.

Furthermore, it is worth noting that the infusion of programming into mathematics education may not always yield positive outcomes, as revealed by research conducted by Dohn (2020). The work of Bråting et al. (2020) pointed out the syntactic and semantic disparities between algebra and mathematics, which can pose challenges to students' algebra learning within the context of programming. This underscores the importance of careful implementation.

Misfeldt and Ejsing-Dunn (2015) raise a cautionary flag regarding the role of teachers in effectively conveying core mathematical concepts within the framework of programming. Hence, teacher competencies in this area should be thoughtfully addressed.

Additionally, future research should place a greater emphasis on collaborative learning, as suggested by Perez (2018), who proposed a framework of computational thinking dispositions supporting students' mathematical knowledge and skill development. Collaboration emerges as a pivotal life skill, yet our review reveals that only a limited number of studies have explored students' collaboration strategies within the context of computational thinking-integrated mathematics curricula.

Finally, the dearth of studies examining collaboration skills among students, as noted by Forsström and Kaufmann (2018), necessitates further exploration into this critical facet of integrated education.

In conclusion, the amalgamation of computational thinking and mathematics education demonstrates substantial potential. However, its successful implementation requires meticulous curriculum design, ongoing teacher professional development, and thoughtful pedagogical practices. Moreover, it prompts a clarion call for additional research to effectively address challenges and discern best practices, thereby facilitating the seamless integration of computational thinking into the realm of mathematics education.

Discussion and Conclusion

The converging insights gleaned from the reviewed literature underscore the multifaceted and profound implications of fusing computational thinking practices with mathematics education. The positive outcomes resonate across various dimensions, notably enhanced mathematical comprehension, problem-solving processes, and creative ideation. The synthesis of these findings reinforces the notion that the inclusion of programming activities can imbue mathematics with renewed accessibility, vibrancy, and relevance in students' lives. As suggested by Darmawansah et al. (2023), the intricate interplay between computational thinking and mathematical proficiency fosters a more holistic understanding of both domains, a notion that can potentially revolutionize conventional pedagogical paradigms.

Furthermore, the reciprocal benefits manifest in the augmentation of students' motivation, interest, and engagement. In line with the previous reviews (Chan et al., 2023; Ye et al., 2023), it is found that the dynamic, interactive nature of programming activities captivates learners' interest and stimulates their cognitive faculties, leading to a heightened sense of ownership and agency over their learning journey. Such a shift from passive recipients to active co-creators of knowledge has the potential to reshape the way mathematics is perceived and internalized, consequently fostering a more enduring and meaningful grasp of mathematical concepts.

Moreover, the resonance of computational thinking with the fundamental goals of mathematics education—critical thinking, logical reasoning, and problem-solving—provides an organic scaffold for embedding programming practices into curricula. While previous reviews focuses mostly either on problem-solving or logical reasoning in mathematics learning (Ye et al., 2023), this review adds critical thinking into the sub-themes together with the two. This symbiotic relationship can potentially mitigate the oft-perceived abstraction and complexity of mathematics, rendering it more tangible, relatable, and palpable for learners across diverse backgrounds.

To conclude, the exploration of the intricate relationship between computational thinking, programming, and mathematics education has illuminated a compelling avenue for enriching students' learning experiences and academic outcomes. This comprehensive review, encompassing an array of research studies, has delved deep into the symbiotic interplay between computational thinking practices and the realm of mathematics education. By synthesizing findings from diverse contexts, methodologies, and pedagogical approaches, this paper underscores the transformative potential of integrating computational thinking into the fabric of mathematics instruction.

1. Future Directions

As the foundation laid by the current literature is both robust and burgeoning, there are several exciting avenues for future exploration. First of all, a longitudinal lens can provide nuanced insights into the longitudinal impact of computational thinking integration on students' mathematical proficiency, ensuring the sustainability and transferability of acquired skills over time. Furthermore, in the wake of emerging digital divides, rigorous research is needed to strategize and implement equitable access to computational resources, ensuring that the benefits of such integration reach all students, irrespective of socioeconomic status. Moreover, the creation of comprehensive and dynamic assessment frameworks that holistically measure both mathematical understanding and computational thinking competencies will be instrumental in gauging the true impact of integration. Last but not least, exploring the potential of integrating computational thinking practices across other subjects could unravel interdisciplinary intersections, thereby fostering holistic skill development and cognitive synergy.

2. Implications

The findings of this comprehensive review extend their tendrils to various stakeholders in the educational ecosystem: Firstly, teachers stand to benefit from tailored professional development initiatives that empower them with the pedagogical acumen and practical strategies needed for seamlessly integrating computational thinking into their mathematics instruction. Secondly, a well-conceived curriculum that seamlessly marries computational thinking with mathematics can bestow students with indispensable skills while revitalizing the pedagogical landscape. Moreover, the recognition of the catalytic role computational thinking plays in modern education demands the design of policies that allocate resources, facilitate training, and promote innovative teaching methods. Lastly, the research gaps and challenges unearthed by this review chart an ambitious agenda for future studies, including comprehensive longitudinal investigations, cross-cultural comparative analyses, and granular explorations into optimal pedagogical practices.

In sum, the confluence of computational thinking, programming, and mathematics education bears the promise of transformative reformulation. As the realms of education continue to evolve in response to the ever-shifting landscape of the digital age, the integration of computational thinking emerges as a beacon of innovation. By capitalizing on this synergy, educators, researchers, and policymakers can collaboratively propel education into an era characterized by dynamic, engaging, and impactful learning experiences. Through this symbiotic integration, mathematics education transcends its conventional boundaries, empowering students to embrace a digital future with confidence, adaptability, and the intellectual agility necessary to thrive in an interconnected world.

ORCID

Ipek Saralar-Aras: https://orcid.org/0000-0002-4942-4408
Yasemin Cicek Schoenberg: https://orcid.org/0009-0008-6744-6601

Conflict of Interest

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by the JKSMEA by waiving the fee for the special issue.

References

Abid, A., Farooq, M. S., & Farooq, U. (2015). A strategy for the design of introductory computer programming course in high school. *Journal of Elementary Education, 25*(1), 145-165.

Acar, İ. G., & Övez, F. D. (2022). The effect of block-based game development activities on the geometry achievement, computational thinking skills and opinions of seventh-grade students. *Journal of Educational Technology and Online Learning, 5*(4), 1106-1121. https://doi.org/10.31681/jetol.1151170

Ackerman, R. (2023). Bird's-eye view of cue integration: Exposing instructional and task design factors which bias problem solvers. *Educational Psychology Review, 35*(2), 1-37. https://doi.org/10.1007/s10648-023-09771-z

Adanır, G. A., Delen, I., & Gulbahar, Y. (2023). Research trends in K-5 computational thinking education: A bibliometric analysis and ideas to move forward. *Education and Information Technologies, 29*, 3589-3614. https://doi.org/10.1007/s10639-023-11974-4

Akpinar, Y., & Aslan, Ü. (2015). Supporting children's learning of probability through video game programming. *Journal of Educational Computing Research, 53*(2), 228-259. https://doi.org/10.1177/0735633115598492

Al-Yakoob, S. M., & Sherali, H. D. (2015). Mathematical models and algorithms for a high school timetabling problem. *Computers & Operations Research, 61*, 56-68. https://doi.org/10.1016/j.cor.2015.02.011

Amuko, S., Miheso, M., & Ndeuthi, S. (2015). Opportunities and challenges: Integration of ICT in teaching and learning mathematics in secondary schools, Nairobi, Kenya. *Journal of Education and Practice, 6*(24), 1-6. https://eric.ed.gov/?id=EJ1078869

Ardito, G., Mosley, P., & Scollins, L. (2014). We, robot: Using robotics to promote collaborative and mathematics learning in a middle school classroom. *Middle Grades Research Journal, 9*(3), 73-88. https://www.infoagepub.com/products/middle-grades-research-journal-9-3

Benton, L., Saunders, P., Kalas, I., Hoyles, C., & Noss, R. (2018). Designing for learning mathematics through programming: A case study of pupils engaging with place value. *International Journal of Child-Computer Interaction, 16*, 68–76. https://doi.org/10.1016/j.ijcci.2017.12.004

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education, 72*, 145–157. https://doi.org/10.1016/j.compedu.2013.10.020

Bers, M. U., González-González, C., & Armas-Torres, M. B. (2019). Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Computers & Education, 138*, 130–145. https://doi.org/10.1016/j.compedu.2019.04.013

Borg, A., Fahlgren, M., & Ruthven, K. (2020). Programming as a mathematical instrument : the implementation of an analytic framework. In A. Donevska-Todorova, E. Faggiano, J. Trgalova, Z. Lavicza, R. Weinhandl, A. Clark-Wilson, & H. G. Weigand (Eds.), *Proceedings of the 10th ERME Topic Conference on Mathematics Education in the Digital Age (MEDA)* (pp. 435-442). Linz, Austria: University of Linz. https://urn.kb.se/resolve?urn=urn:nbn:se:kau:diva-80187

Brancaccio, A., Marchisio, M., Palumbo, C., Pardini, C., Patrucco, A., & Zich, R. (2015). Problem posing and solving: Strategic Italian key action to enhance teaching and learning mathematics and informatics in the high school. *In 2015 IEEE 39th Annual Computer Software and Applications Conference* (Vol. 2, pp. 845–850). IEEE. https://doi.org/10.1109/COMPSAC.2015.126

Bråting, K., & Kilhamn, C. (2021). The integration of programming in Swedish school mathematics: Investigating elementary mathematics textbooks. *Scandinavian Journal of Educational Research, 66*(4), 594–609. https://doi.org/10.1080/00313831.2021.1897879

Bråting, K., Kilhamn, C., & Rolandsson, L. (2020). Integrating programming in Swedish school mathematics: Description of a research project. In Y. Liljekvist, L. B. Boistrup, J. Häggström, L. Mattsson, O. Olande, & Hanna Palmér (Eds.), *Proceedings of MADIF12: The twelfth research seminar of the Swedish Society for Research in Mathematics Education*, 14–15 Jan 2020, Linnaeus University, Växjö, Sweden (pp. 101–110). https://www.diva-portal.org/smash/get/diva2:1432043/FULLTEXT01.pdf

Brown, J. S., Collins, A., & Duguid, P. (1989). Situated cognition and the culture of learning. *Educational Researcher, 18*(1), 32–42. https://doi.org/10.3102/0013189X018001032

Bulut, M., & Ferri, R. B. (2023). A systematic literature review on augmented reality in mathematics education. *European Journal of Science and Mathematics Education, 11*(3), 556–572. https://doi.org/10.30935/scimath/13124

Calder, N. S. (2018). Using scratch to facilitate mathematical thinking. *Waikato Journal of Education, 23*(2), 43–58. https://doi.org/10.15663/wje.v23i2.654

Çam, E., & Kıyıcı, M. (2022). The impact of robotics-assisted programming education on academic success, problem solving skills and motivation. *Journal of Educational Technology and Online Learning, 5*(1), 47–65. https://doi.org/10.31681/jetol.1028825

Chan, S. W., Looi, C. K., Ho, W. K., Huang, W., Seow, P., Wu, L., & Kim, M. S. (2020, November). Computational thinking activities in number patterns: A study in a Singapore secondary school. In H. J. So, (Ed.), *Proceedings of the 28th International Conference on Computers in Education* (Vol. 1, pp. 171–176). Asia-Pacific Society for Computers in Education. https://apsce.net/icce/icce2020/proceedings/paper_146.pdf

Chan, S. W., Looi, C. K., Ho, W. K., & Kim, M. S. (2023). Tools and approaches for integrating computational thinking and mathematics: A scoping review of current empirical studies. *Journal of Educational Computing Research, 60*(8), 2036–2080. https://doi.org/10.1177/07356331221098793

Chung, C. C., Cartwright, C., & Cole, M. (2014). Assessing the impact of an autonomous robotics competition for STEM education. *Journal of STEM Education: Innovations and Research, 15*(2), 24–34.

Çiftci, S., & Bildiren, A. (2020). The effect of coding courses on the cognitive abilities and problem-solving skills of preschool children. *Computer Science Education, 30*(1), 3–21. https://doi.org/10.1080/08993408.2019.1696169

Daher, W. (2022). Students' motivation to learn mathematics in the robotics environment. *Computers in the Schools, 39*(3), 230–251. https://doi.org/10.1080/07380569.2022.2071227

Darmawansah, D., Hwang, G. J., Chen, M. R. A., & Liang, J. C. (2023). Trends and research foci of robotics-based STEM education: A systematic review from diverse angles based on the technology-based learning model. *International Journal of STEM Education, 10*(1), 1–24. https://doi.org/10.1186/s40594-023-00400-3

de Azevedo Cysneiros Filho, G. A., da Silva, N. C., & Morais, B. S. (2021). A review of papers about block programming from the workshop on computing at school. In M. Carmo *(Eds.), Proceedings of the international conference on Educational and New Development* (END 2021) (pp. 111–115). inScience Press. https://doi.org/10.36315/2021end024

Deeva, G., Bogdanova, D., Serral, E., Snoeck, M., & De Weerdt, J. (2021). A review of automated feedback systems for learners: Classification framework, challenges and opportunities. *Computers & Education, 162*, 104094. https://doi.org/10.1016/j.compedu.2020.104094

DeJarnette, A. F. (2018). Students' conceptions of sine and cosine functions when representing periodic motion in a visual programming environment. *Journal for Research in Mathematics Education, 49*(4), 390–423. https://doi.org/10.5951/jresematheduc.49.4.0390

Dickson, B. A., Kotsopoulos, D. & Harris, L. (2022). The use of coding clubs to develop middle-school students' spatial reasoning abilities. *Digital Experiences in Mathematics Education, 8*(1), 50–69. https://doi.org/10.1007/s40751-022-

00099-x

diSessa, A. A. (2018). Computational literacy and "the big picture" concerning computers in mathematics education. *Mathematical Thinking and Learning, 20*(1), 3-31. https://doi.org/10.1080/10986065.2018.1403544

Dohn, N. B. (2020). Students' interest in Scratch coding in lower secondary mathematics. *British Journal of Educational Technology, 51*(1), 71-83. https://doi.org/10.1111/bjet.12759

Duncan, C., & Bell, T. (2015, November). A pilot computer science and programming course for primary school students. *In Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 39-48). Association for Computing Machinery. https://doi.org/10.1145/2818314.2818328

Eglash, R., Bennett, A., Drazan, J., Lachney, M., & Babbitt, W. (2017). A mathematical tool kit for generative justice. *ETD-Educação Temática Digital, 19*(3), 761-785. https://doi.org/10.20396/etd.v19i3.8648374

Eisenberg, M., Basman, A., & Hsi, S. (2014). Math on a sphere: Making use of public displays in mathematics and programming education. *Knowledge Management & E-Learning, 6*(2), 140-155. https://doi.org/10.34105/j.kmel.2014.06.010

Fagerlund, J., Häkkinen, P., Vesisenaho, M., & Viiri, J. (2021). Computational thinking in programming with Scratch in primary schools: A systematic review. *Computer Applications in Engineering Education, 29*(1), 12-28. https://doi.org/10.1002/cae.22255

Felicia, A., & Sharif, S. (2014). A review on educational robotics as assistive tools for learning mathematics and science. *International Journal of Computer Science Trends and Technology, 2*(2), 62-84. https://www.ijcstjournal.org/volume-2/issue-2/IJCST-V2I2P15.pdf

Fields, D., Vasudevan, V., & Kafai, Y. B. (2015). The programmers' collective: Fostering participatory culture by making music videos in a high school Scratch coding workshop. *Interactive Learning Environments, 23*(5), 613-633. https://doi.org/10.1080/10494820.2015.1065892

Foerster, K. T. (2016). Integrating programming into the mathematics curriculum: Combining scratch and geometry in grades 6 and 7. *In Proceedings of the 17th annual conference on information technology education* (pp. 91-96). Association for Computing Machinery. https://doi.org/10.1145/2978192.2978222

Fofang, J. S., Weintrop, D., Walton, M., Elby, A., & Walkoe, J. (2020). Mutually supportive mathematics and computational thinking in a fourth-grade classroom. In M. Gresalfi, & I. S. Horn (Eds.), *The Interdisciplinarity of the Learning Sciences, 14th International Conference of the Learning Sciences* (ICLS) 2020, Volume 3 (pp. 1389-1396). Nashville, Tennessee: International Society of the Learning Sciences. https://repository.isls.org//handle/1/6341

Forsström, S. E., & Kaufmann, O. T. (2018). A literature review exploring the use of programming in mathematics education. *International Journal of Learning, Teaching and Educational Research, 17*(12), 18-32. https://doi.org/10.26803/ijlter.17.12.2

Francis, K., Davis, B. (2018). Coding robots as a source of instantiations for arithmetic. *Digital Experiences in Mathematics Education, 4*, 71-86. https://doi.org/10.1007/s40751-018-0042-7

Francis, K., Khan, S., & Davis, B. (2016). Enactivism, spatial reasoning and coding. *Digital Experiences in Mathematics Education, 2*, 1-20. https://doi.org/10.1007/s40751-015-0010-4

Gadanidis, G., Clements, E., & Yiu, C. (2018). Group theory, computational thinking, and young mathematicians. *Mathematical Thinking and Learning, 20*(1), 32-53. https://doi.org/10.1080/10986065.2018.1403542

Grover, S., Dominguez, X., Leones, T., Kamdar, D., Vahey, P., & Gracely, S. (2022). Strengthening early STEM learning by integrating CT into science and math activities at home. In A. Ottenbreit-Leftwich, & P. Yadav (Eds.), *Computational thinking in preK-5: Empirical Evidence for Integration and Future Directions* (pp. 72-84). Association for Computing Machinery. https://doi.org/10.1145/3507951.3519290

Grover, S., Pea, R., & Cooper, S. (2015). Designing for deeper learning in a blended computer science course for middle school students. *Computer Science Education, 25*(2), 199-237. https://doi.org/10.1080/08993408.2015.1033142

Han, B., Bae, Y., & Park, J. (2016). The effect of mathematics achievement variables on scratch programming activities of elementary school students. *International Journal of Software Engineering and Its Applications, 10*(12), 21-30. https://www.earticle.net/Article/A297536

Hava, K., & Cakir, H. (2017). A systematic review of literature on students as educational computer game designers. In J. Johnston (Ed.), *Proceedings of EdMedia+ Innovate Learning* (pp. 407-419). Association for the Advancement of Computing in Education (AACE). https://www.learntechlib.org/p/178342/

Hickmott, D., Prieto-Rodriguez, E., & Holmes, K. (2018). A scoping review of studies on computational thinking in K-12 mathematics classrooms. *Digital Experiences in Mathematics Education, 4*(1), 48-69. https://doi.org/10.1007/s40751-017-0038-8

Hill, C., Dwyer, H. A., Martinez, T., Harlow, D., & Franklin, D. (2015, February). Floors and Flexibility: Designing a programming environment for 4th-6th grade classrooms. *In Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (pp. 546-551). Association for Computing Machinery. https://doi.org/10.1145/2676723.2677275

Hsu, Y. C., Baldwin, S., & Ching, Y. H. (2017). Learning through making and maker education. *TechTrends, 61*(6), 589-594. https://doi.org/10.1007/s11528-017-0172-6

Huang, W., & Looi, C. K. (2021). A critical review of literature on "unplugged" pedagogies in K-12 computer science and computational thinking education. *Computer Science Education, 31*(1), 83-111. https://doi.org/10.1080/08993408.202

0.1789411
Hudson, M. A., Baek, Y., Ching, Y., & Rice, K. (2020) Using a multifaceted robotics-based intervention to increase student interest in STEM subjects and careers. *Journal for STEM Education Research, 3*(3), 295-316. https://doi.org/10.1007/s41979-020-00032-0
Israel, M., & Lash, T. (2020). From classroom lessons to exploratory learning progressions: Mathematics+ computational thinking. *Interactive Learning Environments, 28*(3), 362-382. https://doi.org/10.1080/10494820.2019.1674879
Jiang, H., Turnbull, D., Wang, X., Chugh, R., Dou, Y., & Chen, S. (2022). How do mathematics interest and self-efficacy influence coding interest and self-efficacy? A structural equation modeling analysis. *International Journal of Educational Research, 115*, 102058. https://doi.org/10.1016/j.ijer.2022.102058
Kado, K. (2022). A teaching and learning the fundamental of calculus through Python-based coding. *International Journal of Didactical Studies, 3*(1), 15006. https://doi.org/10.33902/IJODS.202215006
Kampylis, P., Dagienė, V., Bocconi, S., Chioccariello, A., Engelhardt, K., Stupurienė, G., & Earp, J. (2023). Integrating computational thinking into primary and lower secondary education. *Educational Technology & Society, 26*(2), 99-117. https://doi.org/10.30191/ETS.202304_26(2).0008
Ke, F. (2014). An implementation of design-based learning through creating educational computer games: A case study on mathematics learning during design and computing. *Computers & Education, 73,* 26-39. https://doi.org/10.1016/j.compedu.2013.12.010
Kilhamn, C., Bråting, K., Helenius, O., & Mason, J. (2022). Variables in early algebra: exploring didactic potentials in programming activities. *ZDM-Mathematics Education, 54*(6), 1273-1288. https://doi.org/10.1007/s11858-022-01384-0
Kim, S., & Lee, C. (2016). Effects of robot for teaching geometry to fourth graders. *International Journal of Innovation in Science and Mathematics Education, 24*(2), 52-70. https://core.ac.uk/download/pdf/229407673.pdf
Kim, S. W., & Lee, Y. (2016). The analysis on research trends in programming based STEAM education in Korea. *Indian Journal of Science and Technology, 9*(24), 1-11. https://doi.org/10.17485/ijst/2016/v9i24/96102
Kordaki, M., & Kakavas, P. (2017). Digital tools used for the development of computational thinking in primary education: a ten year systematic literature review. In L. Gómez Chova, A. López Martínez, & I. Candel Torres (Eds.), *Proceedings of EDULEARN 17* (pp. 6346-6356). IATED Academy. https://doi.org/10.21125/edulearn.2017.2437
Krishnamoorthy, S. P., & Kapila, V. (2016, October). Using a visual programming environment and custom robots to learn c programming and K-12 stem concepts. *In Proceedings of the 6th Annual Conference on Creativity and Fabrication in Education* (pp. 41-48). Association for Computing Machinery. https://doi.org/10.1145/3003397.3003403
Lavigne, H. J., Lewis-Presser, A., & Rosenfeld, D. (2020). An exploratory approach for investigating the integration of computational thinking and mathematics for preschool children. *Journal of Digital Learning in Teacher Education, 36*(1), 63-77. https://doi.org/10.1080/21532974.2019.1693940
León, J. M., Robles, G., & Román-González, M. (2016). Code to learn: Where does it belong in the K-12 curriculum? *Journal of Information Technology Education: Research, 15*, 283-303. https://doi.org/10.28945/3521
Leung, H. Y. H. (2021). Systematic literature review of integrating computational thinking into mathematics education. *US-China Education Review, 11*(4), 152-163. https://doi.org/10.17265/2161-623X/2021.04.002
Lin, C. F., Yeh, Y. C., Hung, Y. H., & Chang, R. I. (2013). Data mining for providing a personalized learning path in creativity: An application of decision trees. *Computers & Education, 68*, 199-210. https://doi.org/10.1016/j.compedu.2013.05.009
Liu, J., Feenstra, W., Saenz Otero, A., & Magrane, K. (2014, April). STEM education: Students touch space through free robotics programming competition. In S. Zvcek, M. Teresa Restivo, J. Uhomoibhi, & M. Helfert (Eds.), *Proceedings of the 6th International Conference on Computer Supported Education-Volume 3* (pp. 270-274). SciTePress. https://doi.org/10.5220/0004944802700274
Liu, J., Sun, M., Dong, Y., Xu, F., Sun, X., & Zhou, Y. (2022). The mediating effect of creativity on the relationship between mathematic achievement and programming self-efficacy. *Frontiers in Psychology, 12*, 772093. https:///doi.org/10.3389/fpsyg.2021.772093
Luo, T., Reynolds, J., & Muljana, P. S. (2022). Elementary students learning computer programming: An investigation of their knowledge retention, motivation, and perceptions. *Educational Technology Research and Development, 70*(3), 783-806. https://doi.org/10.1007/s11423-022-10112-0
Mari, E., Millon Faure, K., & Assude, T. (2022). Programmable floor robots and spatial knowledge with 6-7-year-old students. *International Journal for Technology in Mathematics Education, 29*(2), 87-95.
Martin, F., Bacak, J., Polly, D., & Dymes, L. (2023). A systematic review of research on K12 online teaching and learning: Comparison of research from two decades 2000 to 2019. *Journal of Research on Technology in Education, 55*(2), 190-209. https://doi.org/10.1080/15391523.2021.1940396
Miller, J. (2019). STEM education in the primary years to support mathematical thinking: Using coding to identify mathematical structures and patterns. *ZDM, 51*(6), 915-927. https://doi.org/10.1007/s11858-019-01096-y
Misfeldt, M., & Ejsing-Duun, S. (2015, February). Learning mathematics through programming: An instrumental approach to potentials and pitfalls. In K. Krainer & N. Vondrová (Eds.), *Proceedings of the ninth Congress of the European Society for Research in Mathematics Education* (CERME 9) (pp. 2524-2530). Charles University in Prague, Faculty of Education and ERME. https://hal.science/hal-01289367/

Mohamadou, Y., Halidou, A., & Kapen, P. T. (2020). A review of mathematical modeling, artificial intelligence and datasets used in the study, prediction and management of COVID-19. *Applied Intelligence, 50*(11), 3913-3925. https://doi.org/10.1007/s10489-020-01770-9

Monga, M. (2019). Problem posing and programming as a general approach to foster the learning of mathematics. *Constructivist Foundations, 14*(3), 335-337. https://constructivist.info/14/3/335

Montiel, H., & Gomez-Zermeño, M. G. (2021). Educational challenges for computational thinking in K-12 education: A systematic literature review of "scratch" as an innovative programming tool. *Computers, 10*(6), 69. https://doi.org/10.3390/computers10060069

Moore, J. A., & Chow, J. C. (2021). Recent progress and applications of gold nanotechnology in medical biophysics using artificial intelligence and mathematical modeling. *Nano Express, 2*(2), 022001. https://doi.org/10.1088/2632-959X/abddd3

Nemiro, J., Larriva, C., & Jawaharlal, M. (2017). Developing creative behavior in elementary school students with robotics. *The Journal of Creative Behavior, 51*(1), 70-90. https://doi.org/10.1002/jocb.87Citations

Papavlasopoulou, S., Giannakos, M. N., & Jaccheri, L. (2017). Empirical studies on the Maker Movement, a promising approach to learning: A literature review. *Entertainment Computing, 18*, 57-78. https://doi.org/10.1016/j.entcom.2016.09.002

Pei, C., Weintrop, D., & Wilensky, U. (2018). Cultivating computational thinking practices and mathematical habits of mind in lattice land. *Mathematical Thinking and Learning, 20*(1), 75-89. https://doi.org/10.1080/10986065.2018.1403543

Pellas, N., Mystakidis, S., & Christopoulos, A. (2021). A systematic literature review on the user experience design for game-based interventions via 3D virtual worlds in K-12 education. *Multimodal Technologies and Interaction, 5*(6), 28. https://doi.org/10.3390/mti5060028

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

Phan, H. P., Ngu, B. H., & Yeung, A. S. (2017). Achieving optimal best: Instructional efficiency and the use of cognitive load theory in mathematical problem solving. *Educational Psychology Review, 29,* 667-692. https://doi.org/10.1007/s10648-016-9373-3

Popat, S., & Starkey, L. (2019). Learning to code or coding to learn? A systematic review. *Computers & Education, 128*, 365-376. https://doi.org/10.1016/j.compedu.2018.10.005

Psycharis, S., & Kallia, M. (2017). The effects of computer programming on high school students' reasoning skills and mathematical self-efficacy and problem solving. *Instructional Science, 45*(5), 583-602. https://doi.org/10.1007/s11251-017-9421-5

Qian, Y., & Lehman, J. (2017). Students' misconceptions and other difficulties in introductory programming: A literature review. *ACM Transactions on Computing Education (TOCE), 18*(1), 1-24. https://doi.org/10.1145/3077618

Qian, Y., & Lehman, J. D. (2017). Correlates of success in introductory programming: A study with middle school students. *Journal of Education and Learning, 5*(2), 73-83. https://doi.org/10.5539/jel.v5n2p73

Radev, V. (2021). Design of educational computer game in second grade mathematics with the help of scratch. In L. Gómez Chova, A. López Martínez, & I. Candel Torres (Eds.), *Proceedings of EDULEARN 21* (pp. 939-948). IATED Academy. https://doi.org/0.21125/edulearn.2021.0248

Raiyn, J. (2016). Developing a mathematics lesson plan based on visual learning technology. *International Journal of Education and Management Engineering, 6*(4), 1-9. https://doi.org/10.5815/ijeme.2016.04.01

Rich, K. M., Spaepen, E., Strickland, C., & Moran, C. (2020). Synergies and differences in mathematical and computational thinking: Implications for integrated instruction. *Interactive Learning Environments, 28*(3), 272-283. https://psycnet.apa.org/record/2019-25040-001

Rich, P. J., Bly, N., & Leatham, K. R. (2014). Beyond cognitive increase: Investigating the influence of computer programming on perception and application of mathematical skills. *Journal of Computers in Mathematics and Science Teaching, 33*(1), 103-128. https://www.learntechlib.org/p/41965/

Sáez-López, J. M., Sevillano-García, M. L., & Vazquez-Cano, E. (2019). The effect of programming on primary school students' mathematical and scientific understanding: Educational use of mBot. *Educational Technology Research and Development, 67*(6), 1405-1425. https://doi.org/10.1007/s11423-019-09648-5

Saralar-Aras, İ. (2021). *Artificial intelligence applications in mathematics lessons* [Oral Presentation]. The 1st National Conference of Artificial Intelligence Applications in Education, Harran University, Şanlıurfa, Türkiye.

Saralar-Aras, İ., Erkol, E. D., & Akan, R. (2023). Matematik öğretiminde yapay zekâ eğitim uygulamaları [Artificial intelligence education practices for teaching mathematics]. In İ. Saralar-Aras (Ed.) *Maths of artificial Intelligence: Theory and practices* (pp. 49-72). Nobel Publications.

Shute, V. J., & Rahimi, S. (2017). Review of computer-based assessment for learning in elementary and secondary education. *Journal of Computer Assisted Learning, 33*(1), 1-19. https://doi.org/10.1111/jcal.12172

Smith, C. P., & Neumann, M. D. (2014). Scratch it out! Enhancing geometrical understanding. *Teaching Children Mathematics, 21*(3), 185-188. https://doi.org/10.5951/teacchilmath.21.3.0185

Steffe, L. P., & Gale, J. E. (1995). *Constructivism in education*. Psychology Press.

Sümen, Ö. Ö. (2022). Teaching the order of operations topic to fourth-graders using code. org. *Milli Eğitim Dergisi, 51*(236), 3593-3616. https://doi.org/10.37669/milliegitim.970167
Sun, L., Guo, Z., & Hu, L. (2021). Educational games promote the development of students' computational thinking: A meta-analytic review. *Interactive Learning Environments, 21*(1), 1-15. https://doi.org/10.1080/10494820.2021.1931891
Sun, L., Hu, L., & Zhou, D. (2021). Improving 7th-graders' computational thinking skills through unplugged programming activities: A study on the influence of multiple factors. *Thinking Skills and Creativity, 42*, 100926. https://doi.org/10.1016/j.tsc.2021.100926
Sweller, J. (2011). Cognitive load theory. In J. P. Mestre & B. H. Ross (Eds.), *Psychology of Learning and Motivation* (Vol. 55, pp. 37-76). Academic Press. https://doi.org/10.1016/B978-0-12-387691-1.00002-8
Tian, Y., Xiao, J., Jiang, J., Wang, H., & Yuan, Y. (2017). The research on programming education in elementary schools from five countries. *International Journal of Social Media and Interactive Learning Environments, 5*(3), 181-190. https://doi.org/10.1504/IJSMILE.2017.087946
Troelstra, A. S., & Van Dalen, D. (2014). *Constructivism in mathematics* (Vol 2). Elsevier.
Wang, H. Y., Huang, I., & Hwang, G. J. (2014). Effects of an integrated Scratch and project-based learning approach on the learning achievements of gifted students in computer courses. *In 2014 IIAI 3rd International Conference on Advanced Applied Informatics* (pp. 382-387). IEEE. https://doi.org/10.1109/IIAI-AAI.2014.85
Wang, H. Y., Huang, I., & Hwang, G. J. (2016). Comparison of the effects of project-based computer programming activities between mathematics-gifted students and average students. *Journal of Computers in Education, 3*, 33-45. https://doi.org/10.1007/s40692-015-0047-9
Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology, 25*, 127-147. https://doi.org/10.1007/s10956-015-9581-5
Wilson, B. G., & Myers, K. M. (2000). Situated cognition in theoretical and practical context. In D. H. Jonassen & S. M. Land (Eds.), *Theoretical foundations of learning environments* (pp. 57-88). Lawrence Erlbaum.
Ye, H., Liang, B., Ng, O. L., & Chai, C. S. (2023). Integration of computational thinking in K-12 mathematics education: A systematic review on CT-based mathematics instruction and student learning. *International Journal of STEM Education, 10*(1), 3. https://doi.org/10.1186/s40594-023-00396-w
Zhang, L., & Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education, 141*, 103607. https://doi.org/10.1016/j.compedu.2019.103607
Zhang, Y., Luo, R., Zhu, Y., & Yin, Y. (2021). Educational robots improve K-12 students' computational thinking and STEM attitudes: Systematic review. *Journal of Educational Computing Research, 59*(7), 1450-1481. https://doi.org/10.1177/0735633121994070
Zhang, Y., Ng, O. L., & Leung, S. (2023). Researching computational thinking in early childhood STE(A)M education context: A descriptive review on the state of research and future directions. *Journal for STEM Education Research, 6*, 427-455. https://doi.org/10.1007/s41979-023-00097-7