International Journal of Modern Education
IJMOE
EISSN: 2637-0905

Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

INTERNATIONAL JOURNAL OF MODERN EDUCATION (IJMOE)
www.ijmoe.com

GLOBAL ACADEMIC EXCELLENCE

COMPUTATIONAL THINKING AND PAIR PROGRAMMING: A SYSTEMATIC REVIEW

Zuraihah Ngadengon[1], Tamil Selvan Subramaniam[2*], Zurina Yasak[3], Muhammad Syukri[4], Muhamad Noor Hazim[5]

[1] Faculty of Technical and Vocational Education, Universiti Tun Hussein Onn Malaysia, Malaysia
Email: gb210016@student.uthm.edu.my
[2] Faculty of Technical and Vocational Education, Universiti Tun Hussein Onn Malaysia, Malaysia
Email: tselvan@uthm.edu.my
[3] Faculty of Technical and Vocational Education, Universiti Tun Hussein Onn Malaysia, Malaysia
Email: zurina@uthm.edu.my
[4] Physics Education, Faculty of Teacher Training and Education, Universitas Syiah Kuala, Indonesia
Email: syukri.physics@usk.ac.id
[5] AVM Cloud Sdn. Bhd., Malaysia
Email: noorhazimesa@gmail.com
[*] Corresponding Author

Article Info:

Article history:
Received date: 13.11.2023
Revised date: 19.12.2023
Accepted date: 01.02.2024
Published date: 05.03.2024

To cite this document:

Ngadengon, Z., Subramaniam, T. S., Yasak, Z., Syukri, M., & Hazim, M. N. (2024). Computational Thinking And Pair Programming: A Systematic Review. *International Journal of Modern Education, 6* (20), 103-119.

DOI: 10.35631/IJMOE.620008

Abstract:

There has been a growing interest in the combined techniques of Pair Programming (PP) and Computational Thinking (CT) due to its potential in enhancing problem-solving skills and fostering collaborative learning environments. While there is a growing body of literature on CT and PP, a systematic synthesis is essential to identify trends, gaps, and methodological nuances. This review addresses the need for a consolidated analysis beyond individual studies to offer a comprehensive analysis of the effects of PP and CT in educational circumstances. Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines are adhered to by the review to ensure a rigorous and transparent methodology. An advanced searching technique is employed, utilizing CT and PP as keywords, to systematically explore the Scopus as well as Web of Science (WoS) databases. Subsequently, the inclusion and exclusion criteria are applied systematically to identify relevant studies, and a thorough quality assessment is conducted. The results present a synthesized overview of the literature, categorizing findings into two main themes derived from expert validation, which are (1) CT and PP in Primary and Elementary Education and (2) CT and PP in Middle and Secondary Education. In conclusion, this systematic review explores the integration of CT and PP in educational settings, offering valuable insights for

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

educators, researchers, and policymakers, emphasizing the importance of systematic approaches.

Keywords:

Computational Thinking, Pair Programming, Systematic Review, Agile Software

Introduction

In the rapidly evolving landscape of computer science education, two key pedagogical approaches have gained considerable attention and acclaim—Computational Thinking (CT) and Pair Programming (PP). As technology continues to permeate various facets of our lives, the development of computational skills becomes paramount, and educators are faced with the challenge of preparing students for an increasingly complex and digital future. CT requires an array of problem-solving abilities based on computer science fundamentals (Wing, 2017). These skills include algorithmic thinking, abstraction, pattern recognition, and decomposition (Rahman et al., 2020). CT is not limited to courses on computer science; instead, it permeates diverse disciplines, fostering a mindset that transcends the boundaries of traditional programming. As we navigate an era dominated by data, CT equips students with the capacity to systematically evaluate and resolve issues, creating the foundation for a generation adept at addressing complex challenges.

PP, a practice originating from agile software development methodologies, involves two individuals collaboratively working on a single computer (Roque-Hernandez et al., 2021). One becomes the coder and acts as the driver, whereas the other becomes the navigator, providing feedback, suggesting improvements, and ensuring the overall quality of the code (Schulz et al., 2023). In the industry, the time allocated by a software developer or programmer for task execution is distributed as follows: 30% for solo programming, 50% for collaborative work with a single peer, and 20% for collaborative efforts involving two or more peers (Nagappan et al., 2003). PP is celebrated for its ability to enhance learning (Beasley & Johnson, 2022; Iskrenovic-Momcilovic, 2019; Umapathy & Ritzhaupt, 2017; Zhong et al., 2016), improve code quality (Alhalhouli et al., 2017), facilitate knowledge transfer among students and foster scaffolding through the communication process (Demir & Seferoglu, 2021). Beyond its immediate benefits, PP has the potential to create an inclusive and supportive learning environment, breaking down barriers and promoting a sense of shared responsibility for the learning process.

Several studies have conducted a systematic review of CT. Researchers have studied systematic reviews related to plugged applications (Bati, 2022) and unplugged applications to foster CT (Chen et al., 2023; Huang & Looi, 2020). For example, a study by Chen et al. (2023) examining the effectiveness of unplugged activities in promoting CT skills in K-12 education between 2006 and 2022 found that board and card games were the most common unplugged activities. They highlighted the potential of unplugged pedagogy in enhancing CT skills in K-12 education. Some studies conducted a systematic review that explored robot activities to promote the development of CT (Bakala et al., 2021; Funk et al., 2021; K. Yang et al., 2020). A study conducted by Yang et al. (2020) found that six different kinds of robots used in CT experiments are Arduino+scratch, virtual and physical robotics, Bee-bot Robotic, KIBO Robotic, LEGO Robotic, and ultra-low-cost line follower Robotic. LEGO Robotics is the most

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

often utilized robot for students of all ages. The research findings demonstrate that a robot classroom can help students develop their CT.

Meanwhile, the systematic review highlights the significant educational impact of visual programming environments in K-12 education focusing on CT using Scratch (Montiel & Gomez-Zermeño, 2021) and ScratchJr (Stamatios, 2024). Stamatios (2024) reviews 18 studies focusing on the impact of ScratchJr on young children's CT, coding skills, and overall literacy. It concludes that while ScratchJr is not a universal solution, it is a beneficial tool for improving these skills. CT has gotten increased attention in mathematics education. There have been several studies in the literature reporting a systematic review regarding CT in mathematics education (Barcelos et al., 2018; Kaup, 2022; Khoo et al., 2022; Subramaniam et al., 2022; Ye et al., 2023). Ye et al. (2023) reviewed 24 articles on CT-based mathematics instruction and found that geometrized programming and student-centered approaches facilitate productive learning. CT-based learning involves an interactive, cyclical process of mathematical and computational reasoning involving the construction of CT artifacts, anticipating and interpreting outputs, and generating new mathematical knowledge.

There have been several studies in the literature conducted a systematic review of PP (Hanks et al., 2011; Hawlitschek et al., 2022; Salleh et al., 2011; Umapathy & Ritzhaupt, 2017). Study by Salleh et al. (2011) presents evidence on the effectiveness of PP as a pedagogical tool in higher education CS/SE courses. A systematic literature review identified 14 compatibility factors affecting PP's effectiveness. Students' skill level, time spent on programming, and satisfaction were found to be the most significant factors. PP was found to improve students' grades and was most effective when measured by test case success, academic performance, and expert opinion. Hanks et al., (2011) examines the benefits of PP in undergraduate computer science curricula, highlighting its advantages, such as higher success rates, improved retention, enhanced software quality, and improved learning outcomes. It also highlights its benefits for female students and challenges like scheduling and partner compatibility. Meanwhile, a meta-analysis of 18 studies involving 3,308 students found that PP is an effective pedagogical tool for student outcomes in computer programming courses. The findings suggest that PP can improve assignments, exams, passing rates, and affective measures (Umapathy & Ritzhaupt, 2017). Hawlitschek et al.(2022) reported that pairing students is an effective method for improving programming performance, especially for beginners. However, lecturers should plan implementation, monitor learning processes, and offer guidance.

While previous systematic reviews have examined CT and PP independently, there is a need for more integrative analysis that combines these two areas. The identified research gap in this study lies in the need for a comprehensive understanding and synthesis of how PP contributes explicitly to developing and enhancing CT skills. The proposed systematic review aims to bridge this gap by meticulously analyzing how PP as a pedagogical tool can effectively augment CT skills. This investigation is crucial for developing targeted educational strategies and refining curriculum designs, ensuring they align with the most effective teaching methodologies for enhancing CT competencies.

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

Literature Review

Computational Thinking

In the twenty-first century, CT is important, particularly for problem-solving. Seymour Papert inaugurated the concept of CT in 1980, to signify the transformative impact that computers could have on cognitive processes within the realm of mathematical education (Papert, 1980). Jeannette Wing reintroduced the term CT in 2006. CT involves not just knowledge about methods for writing programs but also ways of thinking and finding solutions to emerging problems using fundamental computer science principles such as reading, writing, and arithmetic, which every individual should master (Wing, 2006). Wing (2010) further defines CT as a problem-solving process, where information processing agents can be effectively implemented to solve problems. These information-processing agents refer to anything that follows instructions to complete a task, referring to computers, other digital devices, or humans.

Since the initial attempts to elucidate CT, various scholarly perspectives have emerged to define its essence. Aho (2012) describes CT as a cognitive process dedicated to structuring problems so that the solutions are expressible through algorithmic and computational procedures. Refer to Lee & Cho (2020) CT involves using automated ways to solve problems by identifying and abstracting them. This process relies on the fundamental concepts and principles of computational technology. Meanwhile, Tsai, Liang & and Hsu (2020) categorize CT into domain-specific and domain-general categories. The domain-specific category indicates the specialized knowledge or skills required for systematically solving problems within the specific computer science or computer programming domain. On the other hand, the domain-general category refers to the competencies needed for systematically solving problems in everyday life and across all learning domains, viewing CT as a process of thought.

Pair Programming

Pair programming is identified as one of the strategies for learning programming (Silva et al., 2020; Wei et al., 2021). This method has been implemented in software development to boost programmer productivity. Pair programming is a facet of extreme programming, a software development methodology introduced in the mid-1990s (Roque-Hernandez et al., 2021) and has gained widespread adoption in the programming industry. Team members share their knowledge during the pair programming process by discussing their perspectives on tasks, including problem-solving skills and programming concepts (Yang et al., 2016). Consequently, studies indicate that pair programming is more effective for learning programming compared to solo programming (Iskrenovic-Momcilovic, 2019; Regis Anne & Carolin Jeeva, 2022).

Pair programming involves two students collaborating continuously to solve problems using a computer (Xu et al., 2023). Conventional pair programming paradigms typically involve two learners collaborating close to a singular computing station, jointly developing identical code sequences (Hawlitschek et al., 2022). In contrast, contemporary iterations of pair programming exhibit a localized distribution of programmers, diverging from the traditional co-located format known as distributed pair programming (Satratzemi et al., 2023). During the implementation of pair programming, students play the role of the driver and navigator. The driver writes code by controlling the mouse and keyboard, while another student, the navigator, checks and guides the driver, and they periodically switch roles between driver and navigator (Schulz et al., 2023). Papadakis (2018) conducted a study where students switched roles

International Journal of Modern Education
EISSN: 2637-0905
IJMOE
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008
approximately every 20 minutes to take on the driver and navigator roles. In a different study, the students interchangeably switch between the roles of driver and navigator every 5 minutes (Lewis, 2011; Zhong et al., 2017).

Continuous communication, questioning, and answering between the driver and navigator are essential, and they contribute suggestions on the best ways to solve problems (Nicolescu & Plummer, 2003). The driver and navigator communicate at least every 45–60 seconds (Williams et al., 2002). Researchers employ numerous methods to ascertain pairings for pair programming. Williams et al. (2002) randomly assigned pairs rather than allowing students to choose their partners, and students worked with the same partner. In contrast, the study by Ayub et al. (2019, 2020) conducted an experiment that paired a slow-paced student with a fast-paced student.

CT and PP Implementation in School Settings

Researchers have studied CT and PP in school settings. The study is focused on using PP as a pedagogical approach to foster and enhance CT. In primary school programming Lin & Ke, 2020 examines the instructional procedures and strategies of primary school programming education to foster CT. The result shows that the instructional procedure for cultivating CT in programming teaching generally involves five major stages: review, new knowledge exploration, problem analysis, programming on a computer, and reflection. Instructional strategies to cultivate CT include mind map-supported, unplugged programming, pair programming, and log reflection. Another study Al-Jarrah & Pontelli delineates a distinctive contribution through the integration of virtual PP within Alice, aimed at facilitating CT exposure among middle school students. It articulates the development of an innovative extension to Alice, tailored to enable collaborative programming endeavors among this demographic. This enhanced iteration of Alice incorporates support for virtual PP, thereby permitting the distant sharing of a virtual environment and the allocation of specific roles for educators.

The complexity of PP for elementary students is mitigated through collaboration scripts, as evidenced by a pilot study indicating enhanced collaboration, improved CT skills, and positive student perceptions (Ma et al., 2020). The effectiveness of PP in teaching junior high school students CT is explored through project-based learning and graphical programming which has a positive impact in fostering CT (Liu et al., 2022). Furthermore, (Chang & Tsai, 2018) design a curriculum for computer game-making that involves PP roles, using motion-based touchless games to enhance students' CT skills. PP serves as an effective tactic for fostering CT among students. The study results indicate that PP surpasses solo programming in its efficacy in bolstering and enriching students' grasp of fundamental programming concepts and CT skills and elevating their motivation towards programming (Leow & Huang, 2021).

CT and PP Implementation Across Diverse Demographics

This literature review explores the integration of CT and PP methodologies across various target groups, namely undergraduate students, teachers, and individuals with Attention Deficit Hyperactivity Disorder (ADHD). The effectiveness of combining CT and PP in the educational experience of undergraduate students majoring in science, technology, engineering, and mathematics (STEM) has been investigated in earlier research. The DIVAS project contributes to addressing workforce challenges by training students in Python programming through interventions that include PP exercises (Meysenburg et al., 2018). Moreover, attention is

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008
directed towards professional development initiatives for educators. A notable example involves Computing's PD workshops in 2018-2019, complemented by a virtual conference in 2020, utilizing technology and Snap! PP to enhance teacher engagement and collaboration. This intervention resulted in a noticeable improvement in self-efficacy in the CT teaching process (Jocius et al., 2021). An additional investigation discusses the difficulties of teaching CT to individuals with ADHD. By presenting a case study that employs a three-stage method and incorporating PP, this study highlights the effectiveness of this pedagogical approach in facilitating strong learning abilities among students with ADHD (Da Silva et al., 2020).

Material and Methods

The analysis in this study used the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) technique, which is a well-recognised and accepted standard for performing systematic literature reviews. Publication rules are typically essential in providing writers with guidance on how to evaluate and scrutinise the precision and thoroughness of a review while including pertinent and essential information. The PRISMA framework also emphasises the importance of including randomised trial evaluation surveys in systematic analysis reports for various types of studies (Moher et al., 2009) (Figure 1). The methodology of this research was evaluated using two prominent databases, known as Scopus as well as Web of Science (WoS), due to their strong characteristics. Furthermore, this section presents a comprehensive overview of the four main sub-sections: 1) identification, 2) screening, 3) eligibility, as well as 4) data abstraction and analysis.

Identification

The process of paper selection for this report comprises three key phases within the systematic review methodology. In the first phase, keywords were identified, and associated terms were explored using thesauri, dictionaries, encyclopedias, and prior scholarly inquiries. Following the identification of relevant keywords, search strings were formulated for both the Scopus and WoS databases, as can be seen in Table 1. Based on the search string in Table 1, Scopus and WoS are characterized by their unique organizational frameworks, indexing conventions, and categorical distinctions, leading to a divergence in the results of identical search terms attributable to their respective and distinct classification systems. As a result, 60 papers were managed to be obtained from the databases utilized for this research investigation during the first stage of the systematic review procedure.

Table 1: The Search String

Scopus	TITLE-ABS-KEY ("computational thinking" AND "pair programming") AND (LIMIT-TO (PUBYEAR, 2012) OR LIMIT-TO (PUBYEAR, 2013) OR LIMIT-TO (PUBYEAR, 2014) OR LIMIT-TO (PUBYEAR, 2015) OR LIMIT-TO (PUBYEAR, 2016) OR LIMIT-TO (PUBYEAR, 2017) OR LIMIT-TO (PUBYEAR, 2018) OR LIMIT-TO (PUBYEAR, 2019) OR LIMIT-TO (PUBYEAR, 2020) OR LIMIT-TO (PUBYEAR, 2021) OR LIMIT-TO (PUBYEAR, 2022) OR LIMIT-TO (PUBYEAR, 2023)) AND (LIMIT-TO (DOCTYPE, "ar")) AND (LIMIT-TO (PUBSTAGE, "final")) AND (LIMIT-TO (SRCTYPE, "j")) AND (LIMIT-TO (LANGUAGE, "English"))
WoS	"computational thinking" AND "pair programming" (Topic) and Article (Document Types) and English (Languages) and 2023 or 2022 or 2021 or 2020 or 2019 or 2018 or 2017 or 2016 or 2014 or 2011 (Publication Years)

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

Screening

In this phase, duplicate documents will be systematically eliminated from the list of documents that have been retrieved. The initial screening phase resulted in the exclusion of 39 publications, followed by a subsequent phase involving the detailed examination of 21 papers using particular inclusion as well as exclusion standards listed in Table 2. The primary criterion applied pertained to the nature of the literature, with a focus on research papers as the main resource for relevant suggestions. Supplementary materials not covered in the most recent study included book series, conference proceedings, reviews, meta-synthesis, book series, meta-analyses, books, as well as chapters. Furthermore, only English-language publications were included in the review. It is imperative to emphasise that the methodology was confined to the years 2011 through 2023. The search conducted within WoS revealed that there were no articles published in the years 2012 and 2013. In the end, the duplication criteria led to the exclusion of 5 publications.

Table 2: The Selection Criterion

Criterion	Inclusion	Exclusion
Language	English	Non-English
Timeline	2011 to 2023	< 2011
Literature type	Journal (Article)	Book, Conference, Review
Publication Stage	Final	In Press

Eligibility

In the third phase of the PRISMA technique, known as eligibility, the focus is on delineating the criteria for inclusion and exclusion pertinent to the review, as well as detailing the methodology employed for categorising studies for subsequent synthesis (Moher et al., 2009; Page et al., 2021). In the eligibility phase a compilation of 16 articles was assembled. A meticulous examination of each article's title and core content was conducted to ascertain adherence to inclusion criteria and alignment with the specific research objectives. Consequently, two report was excluded due to their deviation from the study's scope, insignificance in title, and lack of relevance in the abstract concerning the study's objectives, as substantiated by empirical evidence. Consequently, 14 articles remained eligible for comprehensive review, as illustrated in Figure 1.

Data Abstraction and Analysis

Throughout this study, integrative analysis—which includes mixed, qualitative, and quantitative research methods—came to light as a crucial evaluation technique. This research's main goal was to find appropriate topics as well as subtopics. The initial phase involved the systematic collection of data, constituting the foundational step in theme development. Figure 1 visually represents the meticulous examination of 14 publications, where the authors systematically scrutinised assertions and content relevant to the study's topics. Following this, a comprehensive evaluation of significant studies related to CT and PP ensued, encompassing methodologies and research findings. Collaborative efforts among authors facilitated the extraction of themes grounded in the study's context, documented through a log that captured analyses, perspectives, queries, and other insights relevant to data interpretation. To ensure coherence, the authors conducted a comparative analysis of results, addressing any inconsistencies in theme design through internal discussions. In instances of conceptual disagreements, authors engaged in collaborative discussions. The produced themes underwent

International Journal of Modern Education
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

refinements for consistency. Two experts with specialised knowledge in both PP and CT independently conducted examinations to strengthen the finding's validity, ensuring domain validity. The iterative process involved adjustments based on the author's discretion, incorporating feedback and comments from expert evaluations.

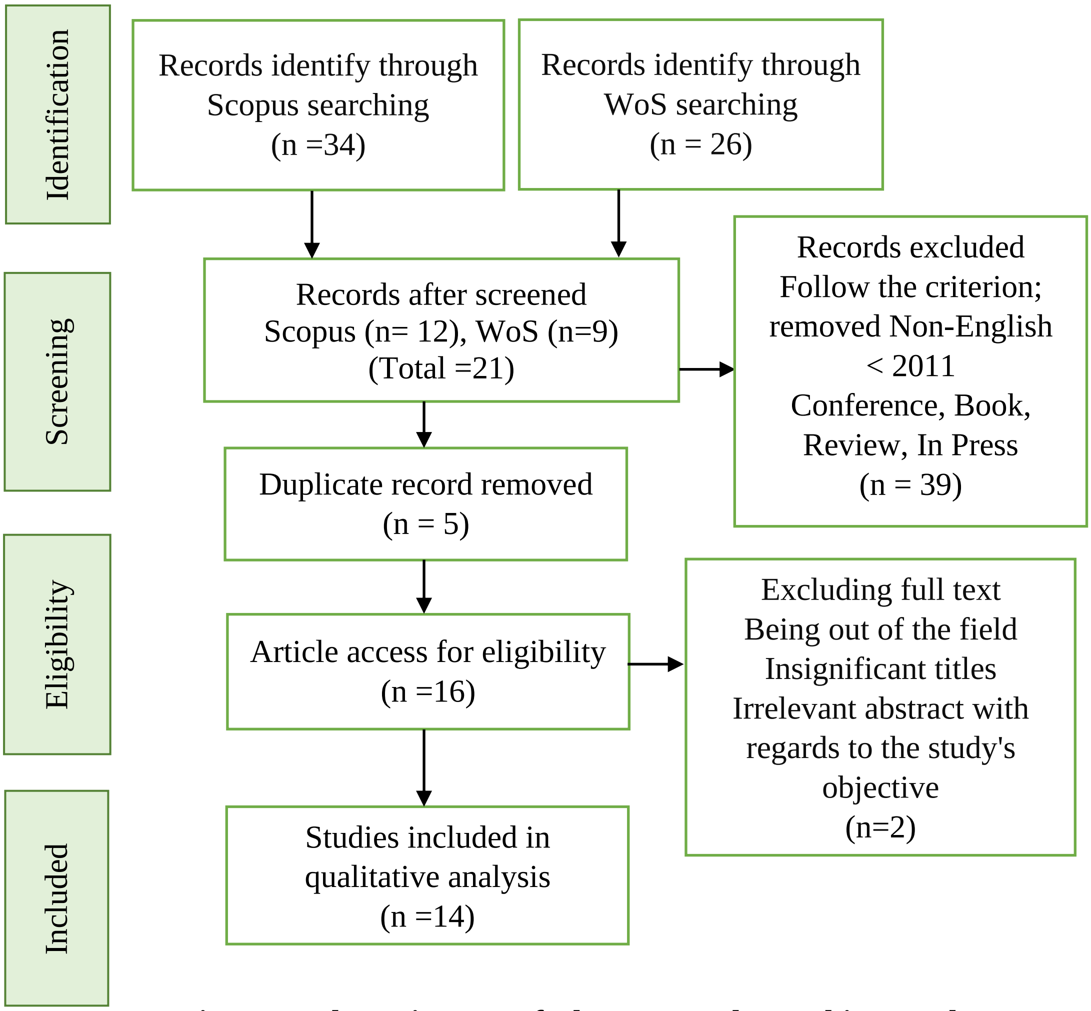

Figure 1: Flow Diagram of The Proposed Searching Study

Source: (Moher et al., 2009)

Result and Finding

The integration of CT through PP involves collaboratively coding with a partner, fostering problem-solving skills, algorithmic reasoning, and effective communication, thereby enhancing students' holistic understanding of computational concepts. Using a systematic search approach, 14 articles were identified and subjected to analysis. These articles were then classified according to two primary themes: (1) PP and CT in Primary and Elementary Education (8 articles) as well as (2) PP and CT in Middle and Secondary Education (6 articles).

Computational Thinking and Pair Programming in Primary and Elementary Education

The collective analysis of these diverse studies reveals significant insights into the pedagogical implications and finding of integrating CT and PP in primary and elementary education. The summary of theme 1 is shown in Table 3.

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

Table 3: Summary of Computational Thinking and Pair Programming in Primary and Elementary Education

Authors	Tools	Country	Aim and finding
Fagerlund J.; Vesisenaho M.; Häkkinen P. (Fagerlund et al., 2022)	Scratch	Finland	This research investigates CT of elementary school students within the framework of collaborative programming in pairs. The findings offer evidence-based pedagogical knowledge for supporting open-ended programming in CT education, including project planning, balancing self-directed design with instructional support, and promoting shared design processes in PP.
Zhong B.; Wang Q.; Chen J.; Li Y. (Zhong et al., 2017)	Scratch	China	The study aimed to investigate the influence of the switching period on PP in promoting CT among young students. The study found that semi-free role switching was more effective for learning achievement than fixed periods, and students who took on a new role every five minutes proved to be more enjoyable. In semi-free classes, role switching became much less common over time, but driver-navigator negotiation ended up being more active.
Hsu T.-C.; Chang C.; Wu L.-K.; Looi C.-K. (Hsu et al., 2022)	Robot	China	The aim is to validate the impact of PP and question-and-response interaction in a board-game activity on young learners' CT skills. The study found that English as a Foreign Language (EFL) students demonstrated better cooperation and problem-solving skills in CT. In contrast, Chinese as a Second Language (CSL) students displayed more trial-and-error behaviours, suggesting cross-disciplinary learning as well as cross-context instruction.
Barth-Cohen L.A.; Jiang S.; Shen J.; Chen G.; Eltoukhy M. (Barth-Cohen et al., 2018)	Robot	United States	The study examines fifth-grade students' CT problem-solving skills in a PP robotics interview. They navigate through various representations, including task instructions, coding interfaces, outputs, and a tangible robot. The findings show proficiency in interpreting and manoeuvring through information simultaneously.
Seo Y.-H.; Kim J.-H. (Seo & Kim, 2016)	Scratch and Entry	Korea	This research investigates the impact of incorporating PP into coding education to enhance CT and creativity among elementary school students. Results showed insignificant differences between the two groups, but the experimental group showed a significant increase in CT and creativity. Using

International Journal of Modern Education
EISSN: 2637-0905
IJMOE
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

			cooperative learning and PP approaches enhances students' CT and creativity.
Wei X.; Lin L.; Meng N.; Tan W.; Kong S.-C.; Kinshuk (Wei et al., 2021)	Scratch	China	This study aimed to evaluate the impact of partial PP (PPP) on elementary school students' CT skills and self-efficacy (SE). In accordance with the research, students within the experiment group outperformed those in the control group in terms of improvement in their CT and SE skills. PP was beneficial for CT and building programming knowledge, especially among less experienced students. Initial computer experience and attitude towards collaboration influenced partners' success.
Bodaker L.; Rosenberg-Kima R.B. (Bodaker & Rosenberg-Kima, 2023)	Scratch	Unknown	The study analyzed elementary school children's performance and attitudes towards an online programming learning activity using the PP method, aiming to enhance collaboration and CT. Children enjoyed online learning activities, with PP benefiting girls. However, they took longer to complete tasks, perceived the third task as harder, and were less active with competent partners.
Zhong B.; Wang Q.; Chen J. (Zhong et al., 2016)	Alice	China	This research aim to investigate how two social variables, namely gender and partnership, influence PP effectiveness in cultivating CT in an elementary school environment. The study found no significant difference in compatibility among gender pairs but significant differences in partnership pairs. Girls showed increased productivity and confidence in PP, suggesting teachers should consider partnerships in collaborative learning to reduce gender gaps and promote socialisation.

Computational Thinking and Pair Programming in Middle and Secondary Education

The collective analysis of these diverse studies reveals significant insights into the pedagogical implications and findings of integrating CT and PP in middle and secondary education. Table 4 displays a summary of theme 2.

Table 4: Summary of Computational Thinking and Pair Programming in Middle and Secondary Education

Authors	Tools	Country	Aim and finding
Denner J.; Werner L.; Campe S.; Ortiz E. (Denner et al., 2014)	Alice	United States	This study examine whether PP is effective for middle school students, what it is effective for, and how partners influence each other. The study reveals that PP is beneficial for CT and programming knowledge, especially among less experienced students, with more experienced

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

			students gaining more confidence and positive attitudes.
Lee J.Y.; Oh S.Y.; Kim S.B. (Lee et al., 2016)	Entry	Korea	The study aims to enhance CT and creativity in coding education through an effective PP scheme, focusing on the advantages of educational programming languages. The study demonstrates that this approach can enhance coding proficiency and mutual learning by fostering collaboration among coding learners.
Werner L.; Denner J.; Campe S.; Torres D.M. (Werner et al., 2020)	Alice	United States	The widespread adoption of CT activities in secondary schools faces a challenge in measuring student learning from game programming and similar activities, known as computational learning (CL). This study refines the Game Computational Sophistication (GCS) model to facilitate the evaluation of computational learning among middle school students engaged in game programming using the PP approach. Finding: (1) Creating GCS 2.0 to generate a singular measure of a game's complexity; (2)Validating this model with 39 games, considering the intricacy added by using multiple game mechanics and (3) Integrating GCS 2.0 into a broader framework for evaluating CL in secondary students that design and programme games.
Sun, D; Ouyang, F; Li, Y; Chen, HY (Sun et al., 2021)	Minecraft	China	The study uses PP to enhance computer programming education in Chinese secondary schools, addressing frustration and boredom among novice programmers in enhancing CT. The results reveal discrepancies among three pairs: low-ranked, middle-ranked, and high-ranked. The low-ranked pair spent more time on distracted activities, while the middle-ranked pair focused on programming explorations and questioning. The high-ranked pair focused more on debugging programming codes. The study also found complex correlations between programming behaviors, discourses, and perceptions, which may significantly influence collaborative programming quality, performance, and experience.
Zhong, BC; Li, TT. (Zhong & Li, 2020)	Robot	China	This research aims to conduct a comparison experiment (pair learning versus individual learning) in robotic education (RE). The study found that students in a pair learning group (PLG) possessed higher success rates in troubleshooting robot artefacts than those in an

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

			individual learning group (ILG) yet were unsuccessful in further indicators. These findings suggest that in the context of robotics education, collaborative (pair) learning is similar to individual learning, particularly in aspects of troubleshooting.
Zha S.; Jin Y.; Moore P.; Gaston J. (Zha et al., 2020)	Hopscotch and Flipped Learning	United States	This study explores the integration of CT using a flipped learning module. The research found that integrating technology and instructional methods like team-based learning, flipped classrooms and PP improved students' understanding and application of CT concepts.

Discussion

Based on the result in Table 3 for Theme 1, the most frequent tool used by PP to foster CT in primary education is Scratch. With its simplicity and visual appeal, Scratch is ideal for introducing basic programming concepts and fostering collaboration. Using a block-based coding system, students can construct computational sequences of instructions (Fagerlund et al., 2022). Furthermore, the three-dimensional framework of CT, encompassing computational concepts, practices, and perspectives, aligns well with the characteristics of Scratch programming. This congruence offers a theoretical foundation for accurately depicting the CT inherent in programming tasks (Jou et al., 2021). From the provided data in Table 3, it is clear that China is the country where such studies have been conducted most frequently, with four out of the eight listed studies taking place there. Finland, the United States, and Korea each have one study, while the location of one study remains unspecified. According to the findings in Table 4, for Theme 2, Alice is the most frequently used tool by PP to promote CT in secondary education. Alice provides visually engaging and simplified programming environments that are suitable for beginners because Alice affords its users the capability to manipulate characters within three-dimensional settings through a drag-and-drop programming interface. (Denner et al., 2014). Table 4's findings indicate that the United States is the country conducting the majority of the studies in this field. This prevalence may reflect a strong interest or initiative in the United States regarding innovative educational strategies like PP in the context of CT education.

The incorporation of CT and PP in educational settings proves highly advantageous. The results demonstrate that PP not only enriches CT and programming knowledge, especially with regard to students with less experience but also instils confidence and positive attitudes in more seasoned learners (Denner et al., 2014). The affirmative impact of PP on CT skills and self-efficacy is conspicuous, especially among less experienced students, underscoring the influence of initial computer experience and collaborative attitudes (Wei et al., 2021). Moreover, the research indicates that integrating technology with traditional teaching methods and PP enhances students' understanding and application of CT concepts (Zha et al., 2020). Although technology use aligns with heightened success rates, it emphasises that collaborative learning may not consistently outperform individual learning across all measures (Zhong & Li, 2020). This underscores the substantial influence of these approaches on students' educational achievements. Online PP activities prove advantageous for girls despite extended task completion times, perceived challenges, and reduced activity levels with proficient partners (Bodaker & Rosenberg-Kima, 2023). Notably, the examination of gender and partnership

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

dynamics suggests the consideration of partnerships in collaborative learning, offering a promising avenue for diminishing gender gaps and fostering socialization (Zhong et al., 2016). The engagement of students in PP not only enhances their technical proficiency but also nurtures a holistic understanding of CT. Moreover, the positive impact of PP extends beyond the academic realms, preparing students for the collaborative demands of the professional world.

Conclusion

Based on the systematic review, the proposed research areas encompass a broad spectrum of investigations into CT and PP for further research. This includes examining the implementation and effects of CT and PP across various educational settings, including tertiary, vocational, and adult education. Additionally, teacher training programs should be evaluated for their effectiveness in implementing CT and PP, and innovative curriculum models should be explored. This strategy enhances technical proficiency and critical thinking skills, aligns with industry demands, and prepares students for modern workplaces. Adequate training and continuous assessment are essential for optimal learning outcomes. Gender dynamics within CT and PP are scrutinized to increase female participation and address gender-specific challenges while examining the impact of these methodologies on social and emotional learning outcomes such as empathy, teamwork, and resilience. Furthermore, the integration of emerging technologies like augmented reality (AR), virtual reality (VR), and artificial intelligence (AI) is explored to enhance educational experiences and advance digital literacy. Finally, there is a demand for the creation of developing PP approach modules emphasizing collaboration, diverse coding activities, role rotation, and problem-solving challenges to augment CT education.

Overall, the systematic review advocates for the widespread adoption of CT and PP practices in primary and secondary education. In conclusion, integrating CT and PP has a positive impact on the students. The reason is that PP has the ability to enhance students' CT in terms of concept and practice. Furthermore, the continuous assessment and refinement of CT through PP implementations are deemed essential, serving as integral components to gauge effectiveness and adapt pedagogical strategies for optimal student learning outcomes. Prospective research and ongoing innovation in this realm are imperative, playing a pivotal role in shaping the future landscape of computer science education and equipping students to effectively confront the evolving challenges of the digital age. This strategic integration not only enhances the quality of learning but also contributes significantly to the formation of a cohort of graduates possessing the intellectual acumen and teamwork proficiency requisite for success in today's intricate and ever-changing work environments. With implications for pedagogy, curriculum development, and gender gap reduction, it provides a robust foundation for educators, curriculum developers, and policymakers to enhance teaching practices and create inclusive learning environments.

Acknowledgements

This research was not funded by any grant.

References

Aho, A. V. (2012). Computation and computational thinking. *Computer Journal*, *55*(7), 832–835. https://doi.org/10.1093/comjnl/bxs074

International Journal of Modern Education IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008
Alhalhouli, Z., Alebeisat, F., Alshabatat, T. E., & Alrawashdeh, T. (2017). Assessing the effectiveness and usability of using pair programming to improve programming language learning, productivity, and code quality. *Journal of Theoretical and Applied Information Technology*, *95*(12), 2783–2791.

Ayub, M., Karnalim, O., Risal, L., & Wijanto, M. C. (2020). The impact of pair programming on the performance of slow-paced students : A study on Data Structure courses. *Journal of Information and Organizational Sciences*, *44*(2), 211–229. https://doi.org/10.31341/jios.44.2.1

Ayub, M., Karnalim, O., Risal, R., Senjaya, W. F., & Wijanto, M. C. (2019). Utilising pair programming to enhance the performance of slow-paced students on Introductory Programming. *Journal of Technology and Science Education*, *9*(3), 357–367. https://doi.org/10.3926/JOTSE.638

Bakala, E., Gerosa, A., Hourcade, J. P., & Tejera, G. (2021). Preschool children, robots, and computational thinking: A systematic review. *International Journal of Child-Computer Interaction*, *29*. https://doi.org/10.1016/j.ijcci.2021.100337

Barcelos, T. S., Munoz, R., Villarroel, R., Merino, E., & Silveira, I. F. (2018). Mathematics learning through computational thinking activities: A systematic literature review. *Journal of Universal Computer Science*, *24*(7), 815–845.

Bati, K. (2022). A systematic literature review regarding computational thinking and programming in early childhood education. *Education and Information Technologies*, *27*, 2059–2082. https://doi.org/10.1007/s10639-021-10700-2

Beasley, Z., & Johnson, A. (2022). The impact of remote pair programming in an upper-level CS course. *Conference on Innovation and Technology in Computer Science Education*, *1*, 235–240. https://doi.org/10.1145/3502718.3524772

Bodaker, L., & Rosenberg-Kima, R. B. (2023). Online pair-programming: elementary school children learning scratch together online. *Journal of Research on Technology in Education*, *55*(5), 799–816. https://doi.org/10.1080/15391523.2022.2036653

Chen, P., Yang, D., Metwally, A. H. S., Lavonen, J., & Wang, X. (2023). Fostering computational thinking through unplugged activities: A systematic literature review and meta-analysis. *International Journal of STEM Education*, *10*(47), 1–25. https://doi.org/10.1186/s40594-023-00434-7

Demir, Ö., & Seferoglu, S. S. (2021). A Comparison of Solo and Pair Programming in Terms of Flow Experience, Coding Quality, and Coding Achievement. *Journal of Educational Computing Research*, *58*(8), 1448–1466. https://doi.org/10.1177/0735633120949788

Denner, J., Werner, L., Campe, S., & Ortiz, E. (2014). Pair programming: Under what conditions is it advantageous for middle school students? *Journal of Research on Technology in Education*, *46*(3), 277–296. https://doi.org/10.1080/15391523.2014.888272

Fagerlund, J., Vesisenaho, M., & Häkkinen, P. (2022). Fourth grade students' computational thinking in pair programming with scratch: A holistic case analysis. *International Journal of Child-Computer Interaction*, *33*, 100511. https://doi.org/10.1016/j.ijcci.2022.100511

Funk, M. G., Cascalho, J. M., Santos, A. I., & Mendes, A. B. (2021). Educational Robotics and Tangible Devices for Promoting Computational Thinking. *Frontiers in Robotics and AI*, *8*, 713416. https://doi.org/10.3389/frobt.2021.713416

Hanks, B., Fitzgerald, S., McCauley, R., Murphy, L., & Zander, C. (2011). Pair programming in education: A literature review. *Computer Science Education*, *21*(2), 135–173. https://doi.org/10.1080/08993408.2011.579808

Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

Hawlitschek, A., Berndt, S., & Schulz, S. (2022). Empirical research on pair programming in higher education: a literature review. *Computer Science Education*, *00*(00), 1–29. https://doi.org/10.1080/08993408.2022.2039504

Huang, W., & Looi, C. K. (2020). 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/10.1080/08993408.2020.1789411

Iskrenovic-Momcilovic, O. (2019). Pair programming with scratch. *Education and Information Technologies*, *24*(5), 2943–2952. https://doi.org/10.1007/s10639-019-09905-3

Jou, M., Chen, P.-C., & Wang, J. (2021). The developmental characteristics of computational thinking and its relationship with technical skills: taking the department of engineering as an example. *Interactive Learning Environments*, *31*(6), 3380–3395. https://doi.org/10.1080/10494820.2021.1928236

Kaup, C. F. (2022). Mapping the relations between computational thinking and mathematics in terms of problem-solving. *Acta Didactica Norden*, *16*(4), 17. https://doi.org/10.5617/adno.9185

Khoo, N. A. K. A. F., Ishak, N. A. H. N., Osman, S., Ismail, N., & Kurniati, D. (2022). Computational thinking in mathematics education: A systematic review. *AIP Conference Proceedings*. https://doi.org/10.1063/5.0102618

Lee, Y., & Cho, J. (2020). Knowledge representation for computational thinking using knowledge discovery computing. *Information Technology and Management*, *21*(1), 15–28. https://doi.org/10.1007/s10799-019-00299-9

Leow, W. M. F., & Huang, W. (2021). Exploring the effectiveness of pair programming in developing students' computational thinking skills through Scratch. *Proceedings of the 5th APSCE International Computational Thinking and STEM in Education Conference 2021*, 2–7.

Lewis, C. M. (2011). Is pair programming more effective than other forms of collaboration for young students? *Computer Science Education*, *21*(2). https://doi.org/10.1080/08993408.2011.579805

Moher, D., Liberati, A., Tetzlaff, J., Altman, D. G., Antes, G., Atkins, D., Barbour, V., Barrowman, N., Berlin, J. A., Clark, J., Clarke, M., Cook, D., D'Amico, R., Deeks, J. J., Devereaux, P. J., Dickersin, K., Egger, M., Ernst, E., Gøtzsche, P. C., … Tugwell, P. (2009). Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. *PLoS Medicine*, *6*, 1–6. https://doi.org/10.1371/journal.pmed.1000097

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

Nagappan, N., Williams, L., Ferzli, M., Wiebe, E., Yang, K., Miller, C., & Balik, S. (2003). Improving the CS1 experience with pair programming. *ACM SIGCSE Bulletin*, *35*(1), 359–362. https://doi.org/10.1145/792548.612006

Nicolescu, R., & Plummer, R. (2003). A pair programming experiment in a large computing course. *Romanian Journal of Information Science and Technology*, *6*, 199–216. http://www.citr.auckland.ac.nz/

Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., … Moher, D. (2021). The PRISMA 2020 statement: an updated

Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008
guideline for reporting systematic reviews. *Systematic Reviews*, *10*(1), 89. https://doi.org/10.1186/s13643-021-01626-4

Papadakis, S. (2018). Is pair programming more effective than solo programming for secondary education novice programmers? A case study. *International Journal of Web-Based Learning and Teaching Technologies*, *13*(1), 1–16. https://doi.org/10.4018/IJWLTT.2018010101

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

Rahman, M. M., Sharker, M. H., & Paudel, R. (2020). An effective approach to teach an introductory computer science course with computational thinking and flow-chart based visual programming. *2020 IEEE Frontiers in Education Conference (FIE)*, 1–7.

Regis Anne, W., & Carolin Jeeva, S. (2022). Empirical Analysis of Pair Programming Using Bloom's Taxonomy and Programmer Rankers Algorithm to Improve the Software Metrics in Agile Development. *International Journal of Software Innovation*, *10*(1), 1–15. https://doi.org/10.4018/IJSI.297624

Roque-Hernandez, R. V., Guerra-Moya, S. A., & Caballero-Rico, F. C. (2021). Acceptance and assessment in student pair-programming: A case study. *International Journal of Emerging Technologies in Learning*, *16*(9), 4–19. https://doi.org/10.3991/ijet.v16i09.18693

Salleh, N., Mendes, E., & Grundy, J. C. (2011). Empirical studies of pair programming for CS/SE teaching in higher education: A systematic literature review. *IEEE Transactions on Software Engineering*, *37*(4), 509–525. https://doi.org/10.1109/TSE.2010.59

Satratzemi, M., Stelios, X., & Tsompanoudi, D. (2023). Distributed Pair Programming in Higher Education: A Systematic Literature Review. *Journal of Educational Computing Research*, *61*(3), 546–577. https://doi.org/10.1177/07356331221122884

Schulz, S., Berndt, S., & Hawlitschek, A. (2023). Exploring students' and lecturers' views on collaboration and cooperation in computer science courses - a qualitative analysis. *Computer Science Education*, *33*(3), 318–341. https://doi.org/10.1080/08993408.2021.2022361

Silva, L., Mendes, A. J., & Gomes, A. (2020). Computer-supported collaborative learning in programming education: A systematic literature review. *IEEE Global Engineering Education Conference, EDUCON*, 1086–1095. https://doi.org/10.1109/EDUCON45650.2020.9125237

Stamatios, P. (2024). Can Preschoolers Learn Computational Thinking and Coding Skills with ScratchJr? A Systematic Literature Review. *International Journal of Educational Reform*, *33*(1), 28–61. https://doi.org/10.1177/10567879221076077

Subramaniam, S., Maat, S. M., & Mahmud, M. S. (2022). Computational thinking in mathematics education: A systematic review. *Cypriot Journal of Educational Sciences*, *17*(6), 2029–2044. https://doi.org/10.18844/cjes.v17i6.7494

Tsai, M.-J., Liang, J.-C., & Hsu, C.-Y. (2020). The computational thinking scale for computer literacy education. *Journal of Educational Computing Research*, *0*(0), 1–24. https://doi.org/10.1177/0735633120972356

Umapathy, K., & Ritzhaupt, A. D. (2017). A Meta-Analysis of Pair-Programming in Computer Programming Courses : Implications for Educational Practice. *ACM Transactions on Computing Education*, *17*(4), 1–13. https://doi.org/10.1145/2996201

Wei, X., Lin, L., Meng, N., Tan, W., Kong, S. C., & Kinshuk. (2021). The effectiveness of partial pair programming on elementary school students' Computational Thinking skills

International Journal of Modern Education
IJMOE
EISSN: 2637-0905
Volume 6 Issue 20 (March 2024) PP. 103-119
DOI: 10.35631/IJMOE.620008

and self-efficacy. *Computers and Education*, *160*, 104023. https://doi.org/10.1016/j.compedu.2020.104023

Williams, L., Wiebe, E., Yang, K., Ferzli, M., & Miller, C. (2002). In support of pair programming in the introductory computer science course. *Computer Science Education*, *12*(3), 197–212. https://doi.org/10.1076/csed.12.3.197.8618

Wing, J. M. (2006). Computational Thinking. *Communications of The ACM*, *49*(3), 33–35. https://doi.org/10.1201/b16812-43

Wing, J. M. (2010). *Computational Thinking: What and Why?* Thelink - The Magaizne of the Varnegie Mellon University School of Computer Science. https://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf

Wing, J. M. (2017). Computational thinking's influence on research and education for all. *Italian Journal of Educational Technology*, *25*(2), 7–14. https://doi.org/10.17471/2499-4324/922

Xu, W., Wu, Y., & Ouyang, F. (2023). Multimodal learning analytics of collaborative patterns during pair programming in higher education. *International Journal of Educational Technology in Higher Education*, *20*(1). https://doi.org/10.1186/s41239-022-00377-z

Yang, K., Liu, X., & Chen, G. (2020). The influence of robots on students‟ computational thinking: A literature review. *International Journal of Information and Education Technology*, *10*(8), 627–631. https://doi.org/10.18178/ijiet.2020.10.8.1435

Yang, Y. F., Lee, C. I., & Chang, C. K. (2016). Learning motivation and retention effects of pair programming in data structures courses. *Education for Information*, *32*(3), 249–267. https://doi.org/10.3233/EFI-160976

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). https://doi.org/10.1186/s40594-023-00396-w

Zha, S., Jin, Y., Moore, P., & Gaston, J. (2020). Hopscotch into Coding: Introducing Pre-Service Teachers Computational Thinking. *TechTrends*, *64*(1). https://doi.org/10.1007/s11528-019-00423-0

Zhong, B., & Li, T. (2020). Can Pair Learning Improve Students' Troubleshooting Performance in Robotics Education? *Journal of Educational Computing Research*, *0*(0), 1–29. https://doi.org/10.1177/0735633119829191

Zhong, B., Wang, Q., & Chen, J. (2016). The impact of social factors on pair programming in a primary school. *Computers in Human Behavior*, *64*, 423–431. https://doi.org/10.1016/j.chb.2016.07.017

Zhong, B., Wang, Q., Chen, J., & Li, Y. (2017). Investigating the period of switching roles in pair programming in a primary school. *Educational Technology and Society*, *20*(3), 220–233.