



Computational Thinking: Moving Towards Skill- Based Learning in Mathematics Class

Ritu Singh^{1*}, Ajeet Kumar Rai²

¹Faculty of Education, Banaras Hindu University, Varanasi, India
ritubhmed19@gmail.com

²Faculty of Education, Banaras Hindu University, Varanasi, India
ajeetrai04@gmail.com

Abstract— The contemporary era of 21st century requires a workforce prepared not just with fundamental knowledge but also with versatile problem-solving skills. Computational thinking (CT) is gradually being incorporated into school curricula globally and it is also considered as a 21st century skill. Computational thinking is defined as a cognitive approach to problem solving that fosters creativity, critical thinking and reasoning ability in students. In mathematics education, traditional approaches often emphasize rote memorization and procedural execution. This paper discusses about a transformative change in education towards skill-based learning. This shift towards skill-based learning is supported by the integration of computational thinking (CT) into the mathematics curriculum. The core components of Computational Thinking (CT), such as decomposition, pattern recognition, abstraction, and algorithm design, offers a comprehensive theoretical foundation or conceptual framework for students to deal with the mathematical problems in a systematic and creative manner. Computational Thinking (CT) is analytical thinking and includes actions like logically organizing and analyzing data. Also, it includes representing data with models and simulations, determining possible solutions, and transforming the generalized solutions to other problem situations. Comprehensive understanding, logical reasoning and novel problem-solving skills can be developed in students by promoting CT skills through mathematics education. In this paper theoretical foundations of integrating computational thinking into mathematics and effective pedagogical strategies for its implementation are explained. It also brings attention to potential benefits of CT integration for students in developing the skills required for success in both academic and professional life. This review aims to provide an overview of the existing literature on CT activities at different levels in mathematics education. Also, to articulate how it is integrated into the teaching and learning of Mathematics. Our findings reveal that CT-integrated mathematics curriculum enhances mathematical achievement, conceptual understanding, problem-solving ability, creativity, reasoning ability, and in-class engagement. The prominent strategies used for CT integration in mathematics are problem-based and project-based learning, unplugged activities without the use of any digital tool, which include hands-on, paper-and-pencil, or verbal exchanges to simulate pivotal programming concepts like algorithm design, problem solving, abstract thinking, and logical reasoning and along with plugged-in activities like, gaming block-based programming, coding, scratch-based activities were also used by the researchers.

Keywords— *Computational Thinking (CT), Skill-based learning, Mathematics Class*

I. INTRODUCTION

The modern landscape of education in the 21st century is undergoing a drastic transition, propelled by rapid advancements in technologies and the evolving demands of the workforce around the world. This shift in the educational landscape entails a reassessment of educational goals, shifting away from the traditional way of learning, which concentrates on rote memorization, to skill-based learning, to endow learners with versatile problem-solving skills. Computational Thinking (CT) has emerged as a crucial skill for this new era, is now viewed as a twenty-first-century skill that refers to a way of problem-solving (NEP, 2020; NCFSE, 2023). Wing's (2006) foundational work uplifted Computational thinking to a foundational analytical ability, aligning it with reading, writing, and arithmetic, and highlighting its global relevance across different disciplines. Review of literature shows that CT has also been explored and integrated with various disciplines like science (Aytekin & Topcu, 2024; Ogegbo & Ramnarain, 2022), mathematics (Acar & Dikkartin Övez, 2022; Çelik & Sevgi, 2025; Rodríguez-Martínez et al., 2020), STEM (Han, 2023; Mariana & Kristanto, 2023), etc.

Contemporaneously, there is a growing demand for skill-based learning (SBL) as an educational paradigm because it is not a way of mere knowledge transmission, but it leads to the acquisition of practical, directly applicable competencies. The synergistic integration of CT and SBL is a critical response to the indispensable future readiness due to its ability to apply theoretical knowledge in practical scenarios. SBL focuses on the acquisition of knowledge through practice and its application, and CT is defined as a foundational way of solving problems. CT, therefore, fulfills the need of students to acquire problem-solving skills. Also, the CT integrated curriculum enhances student' active engagement, which helps in developing skills vital for their future employability. A curriculum that is both skill-based and CT-integrated is not solely an educational advancement but a societal

necessity for fostering competent and adaptable individuals who will be capable of fulfilling the dynamic needs of the future workforce.

In mathematics teaching and learning, traditional approaches often emphasize procedural execution and rote memorization. Studies related to Computational thinking shows that CT- integrated learning is applied at different levels, such as primary (Aytekin & Topcu, 2024; Brackmann et al., 2017; Fiş Erümit, 2024), elementary (Bartolini Bussi & Baccaglini-Frank, 2015; Hu & Wang, 2024; Piedade & Dorotea, 2023), middle (Acar & Dikkartin Övez, 2022; Çelik & Sevgi, 2025; Delal & Oner, 2020; Pan et al., 2024), and secondary (Yılmaz & İzmirli, 2023), etc. Moreover, integration of CT has revealed that it has a positive correlation with mathematical thinking (Wu & Yang, 2022), critical thinking (Mariana & Kristanto, 2023; Wu & Yang, 2022), creativity (Mariana & Kristanto, 2023; Wu & Yang, 2022), problem-solving (Fauzi et al., 2024), mathematical understanding (Kaup et al., 2023), achievement in mathematics (Acar & Dikkartin Övez, 2022), self-efficacy (Pan et al., 2024), in-class engagement and above all it enriches the students' CT skills along the various dimensions of CT.

A majority of the studies have used sophisticated pedagogical interventions involving the use of plugged activities (Acar & Dikkartin Övez, 2022; Büscher, 2025; Çelik & Sevgi, 2025; Harlow et al., 2015; Pan et al., 2024; Piedade & Dorotea, 2023; Rodriguez-Martinez et al., 2020; Subramaniam et al., 2022) such as robotics, programming like scratch-based, block-based programming, etc., with a few studies having their focus on simple and accessible interventions without the involvement of a technology-enriched classroom setup. The unplugged activities were also reported to enhance the students' creative problem-solving skills and enrich their mathematical understanding (Aytekin & Topcu, 2024). Over and above, the studies are largely conducted in a foreign context, with very sparse studies in the Indian context (Iyer, 2019), clearly indicating the knowledge gap. Thus, the proposed review highlights this knowledge gap.

The objective of this review paper is:

1. To present an account of the significance of Skill-Based Learning (SBL).
2. To describe Computational Thinking (CT) and its significance in mathematics
3. To explore the theoretical foundations of integrating Computational Thinking (CT) in skill-based mathematics classrooms.
4. To discuss the effective pedagogical strategies for the implementation of CT-integrated mathematics lessons in school education
5. To discuss the challenges in implementing CT-integrated activities in the mathematics classroom in the context of developing South Asian Countries.

II. METHODOLOGY

A Systematic review technique was used. A systematic review specifies the specific research objectives followed by a search and collection of relevant literature pertinent to the objectives using keywords. The key words used in this review were 'computational thinking*', 'algorithmic thinking*', 'procedural thinking*', 'skill-based learning*', 'skill-based education*', 'competency-based education*', 'mathematics*', 'maths*', 'mathematics education*' 'problem-solving instruction*', 'critical thinking*', 'individualized math*' and also some of the research articles were selected using the intersection of the above keywords like ("computational thinking" OR "algorithmic thinking") AND (math* OR "mathematics education") AND ("skill-based learning" OR "competency-based education") from different databases like ERIC, Google Scholar, SCOPUS, Web of Science, PsycInfo. Among the available literature, articles that were not aligned with the paper's objectives were excluded after reading the title and the abstract of the study. The National Education Policy 2020 document (NEP 2020) and National Curriculum Framework for School Education (NCFSE 2023) were also consulted were additionally included to clarify the objectives of the review.

III. REVIEW FINDINGS

A. *Skill-Based Learning and its Significance in Mathematics*

Skill-based learning (SBL), also known as skill-based training or education, is a pedagogical approach that emphasizes the development of directly applicable, practical skills for specific professions or tasks (Bhasme, 2024). This approach is fundamentally defined as "the acquisition of knowledge through practice and application". It emphasizes active engagement and hands-on experience, guiding students towards mastery in their chosen fields and enabling them to apply in a real-world context (Williams, 2024).

The SBL approach is in strong contrast to the traditional approach of knowledge-based learning, which has historically been the dominant way of learning in academic settings. Traditional approaches mainly focus on the dissemination of information, often through discussions and lectures. It emphasizes comprehension instead of the

practical application of that information (Williams, 2024). In this traditional method, students are often trained to memorize the learning content without acquiring the knowledge, tools, and skills required for real-world jobs or tasks. SBL, conversely, shifts the focus of education to practical activities, exercises, operational experience, and the observable implementation of ideas (Raj, 2024). It shifts away from passive reception of knowledge to active engagement and observable skill development.

Skill-based learning is distinguished by various key characteristics that foster its effectiveness in preparing students for real-world challenges. (Bhasme, 2024). It provides a hands-on learning experience and a practical orientation to real-life situations. SBL intrinsically promotes active participation in learning activities, motivating students to gain hands-on experience by applying their skills. This approach promotes learners to master their skills (Williams, 2024; Raj, 2024). SBL aligns with the demands of the present-day vocational sphere. It prepares learners for the evolving vocational sphere by focusing on skills that enhance their employability (Bhasme, 2024; Raj, 2024; Williams, 2024). By focusing on skill-building, SBL facilitates active learning and deep engagement, which helps students build a lasting understanding compared to passive learning methods by retaining information more effectively (Williams, 2024; Cronin, 2025). Along with academic skills, it promotes holistic development, encompassing pre-academic skills, cognitive abilities, and overall outcomes in areas such as social-emotional growth, language, and literacy (Bhasme, 2024). SBL fosters problem-solving skills and critical thinking. SBL fosters essential 21st-century skills, including creativity, critical thinking, collaboration, communication, resilience, and adaptability (Bhasme, 2024). As learners attain practical proficiencies and apply their knowledge, their self-efficacy and confidence are boosted. SBL also develops habits of continuous improvement and learning, preparing students for a continuously evolving professional sphere (Bhasme, 2024; Raj, 2024).

B. Computational Thinking and its Significance in Mathematics

Computational thinking (CT) is now considered an essential 21st-century skill (Haseski et al., 2018; Weintrop et al., 2021) and is being integrated into curricula of schools worldwide. Although there is still no clear agreement on its precise definition, the concept of computational thinking has been associated with computer science for decades. In the 1950s and 1960s, it was known as “algorithmic thinking,” which means a mindset to formulate conversion problems of some input to an output and then looking for algorithms to perform the conversions. Today, the term has been amplified to include thinking with many abstraction levels, to use mathematics to develop algorithms, and to evaluate how well a solution adapts across different sizes of problems (Denning, 2009). Computational thinking (CT) is defined as a way of problem-solving (Cansu & Cansu, 2019). Computational thinking (CT) at its core is a process of problem-solving that relies on the concepts from computer science (Sheldon, 2017). Wing (2006) describes CT as a way of problem-solving, systems designing, and analyzing human behavior by utilizing the concepts of computer science. In the context of education, it represents an array of problem-solving strategies that involve describing problems and their solutions in a way that an information processing agent, such as a computer, could effectively execute (Team, 2024).

1) Core Components of Computational Thinking

The Review of a vast literature on Computational thinking identified that CT has four core components, namely Decomposition, Pattern Recognition, Abstraction and Algorithmic Thinking (Barcelos et al., 2018; Cansu & Cansu, 2019; Hansen & Hadjerrouit, 2021; Mendrofa, 2024; Sheldon, 2017; Team, 2024) which are defined as follows:

a) Decomposition: Breaking down of complex problems into simplified, easy to understand, and more manageable segments is defined as Decomposition. This skill of Decomposition helps students to simplify complex problems, enabling them to access problems more easily (Rahmawati et al., 2024). Mathematical problems related to place value and addition through expansion in denotes the examples of decomposition. When mathematics learners break a number into its thousands, hundreds, tens, and ones like decomposing 4897 as $4000+800+90+7$, and also in addition, they solve questions like $31+14$ by breaking it into $(31+10) + 4$, which strengthens their mental calculation power (Sheldon, 2017). It is also useful for understanding the problems like regrouping in subtraction and dissection of complex geometric figures (Barcelos et al., 2018).

b) Pattern Recognition: The competence to identify trends, patterns, similarities, or recurrent frameworks within the data or problems is defined as pattern recognition. It helps to generalize problem-solving processes, and also to formulate solutions. In mathematics, pattern recognition plays a vital role for understanding patterns in data like trends in graph, sequence patterns Fibonacci sequences in which each term is the sum of previous two terms. It also helps in identifying symmetry in shapes like mirror line or rotational symmetry and algebraic patterns like $(a+b)^2 = a^2 + b^2 + 2ab$ hold true for all numbers (Sheldon, 2017).

- c) *Abstraction:* Abstraction is a process that involves removing unnecessary details to arrive at a recognizable solution or to demonstrate an intricate system with a easily understandable representation or simplified model (Sheldon, 2017). This critical skill of abstraction guides students to decide about the relevance of information, what is necessary and what can be removed, which enables them to focus on the necessary details or components of a problem (Barcelos et al., 2018). Counting of numbers in mathematics illustrates abstraction, for example when students are able to understand and explain that 32 tables are equal to 32 books in quantity, irrespective of its physical characteristics or attributes like its shape and size (Hansen & Hadjerrouit, 2021).
- d) *Algorithmic Thinking:* It includes following or creating a clear series of steps to achieve a specific goal, resulting in dependable and predictable outcomes (Sheldon, 2017). Algorithmic thinking requires learners to solve the problem not just for an answer, but for a replicable procedure to reach that answer or complete a task. Plotting ordered pairs on a coordinate plane to imitate an artwork (Checovich & Licht, 2022), using standardized procedures for arithmetic operations (Team, 2024), or using a structured series of steps to solve a given problem (Hansen & Hadjerrouit, 2021) are some of the examples of algorithmic thinking in mathematics.

Apart from these four core components, CT has other components such as logic, parallel thinking, recursive thinking, and debugging (Kotsopoulos et al., 2017; Rahmawati et al., 2024). Table I presents the core computational thinking concepts and their application as follows

TABLE I. CORE COMPUTATIONAL THINKING COMPONENTS AND MATHEMATICAL APPLICATIONS

CT Component	Definition (CT Context)	Mathematical Application/Example
Decomposition	Breaking down of complex problems into simplified, easy to understand and more manageable segments (Cummins, 2023; Sheldon, 2017).	Decomposition of numbers in place value determination in the mathematical problems, $235=200+30+5$. Regrouping of numbers in addition and subtraction. Dissection of complex geometric shapes.
Pattern Recognition	The competence to identify trends, patterns, similarities or recurrent frameworks within the data or problems (Cummins, 2023; Sheldon, 2017).	Understanding the patterns of statistical data, trends in graph Fibonacci sequences. Identification of symmetry in shapes and algebraic patterns.
Abstraction	Removal of the unnecessary details to get the required solution or to establish a system with simplified model. (Cummins, 2023; Sheldon, 2017).	Counting of numbers, for example understanding 43 pens are equal to 43 pencils irrespective of its shape and size. Portraying of real-life situations through models or symbols.
Algorithmic Thinking	It is the process of establishing a step-by-step solution of of any problem. (Cummins, 2023; Sheldon, 2017).	Developing a step-by-step solution of the mathematical problem, Data analysis procedure design, Developing standard procedures for arithmetic operations, like long division. Plotting ordered pairs on a coordinate plane to create a particular shape.

2) *Benefits of Integrating CT for Enhanced Mathematical Skill Development and Student Engagement*

There is strong empirical evidence supporting the benefits of CT integration. Researches reveal that integrating CT improves many areas, including critical thinking, problem-solving, and logical reasoning (Cummins, 2023). Along with these cognitive benefits, CT integration has been found to enhance noncognitive skills such as student engagement (Checovich & Licht, 2022), collaboration, creativity, confidence, and motivation (Hunsaker, 2022). These merits of CT facilitate students to be more skilled and handle the challenges of the modern world which is technocentric (Shute et al., 2017).

The Computational Thinking (CT) integrated teaching in mathematics classrooms results in variety of merits, such as enhancing students' mathematical skill, intensified engagement of students (Checovich & Licht, 2022) and more extensive conceptual understanding (Kaup et al., 2023). CT integration in mathematics fosters crucial skills such as decomposition, pattern recognition, abstraction, and algorithm design, which are necessary for developing abstract and analytical thinking abilities (Mendrofa, 2024). It also cultivates the critical thinking skills (Saidin et al., 2021), problem-solving skills (Anwar et al., 2024), and the reasoning ability (Ferreira et al., 2023) of the students. It facilitates students to manage complex problems creatively and systematically, facilitating them to break down problems into manageable parts, identify patterns, and develop step-by-step solutions (Saidin et al., 2021). CT skills are meticulously linked to mathematical thinking (Anwar et al., 2024). Engaging with mathematical concepts through CT activities in an active way helps students to memorize facts more effectively and establish a comprehensive understanding, going beyond the shallow learning (Cronin, 2025). CT integrated

teaching learning enables students to think abstractly, develop conceptual connections, and express their ideas in a more competent way while solving the problem (Dahshan & Galanti, 2024). Integrating CT into mathematics reshape the focus of mathematics learning towards reasoning and sense-making, and these are key factors that lead to strong students' engagement (Checovich & Licht, 2022). By linking real life experiences of students with mathematical problems, their knowledge acquisition becomes more meaningful and relevant (Checovich & Licht, 2022). A positive correlation is found between students' success in mathematics and their CT skills after the review of various studies (Suparman et al., 2025). Also, pedagogical approaches of Computational thinking, like game-based learning contributes to students' engagement (Saidin et al., 2021).

C. Theoretical Frameworks Supporting CT Integration

The theoretical frameworks lay the epistemic and pedagogical groundwork necessary for promoting the development of practical skills and comprehensive understanding. Theories of education, especially constructivism and problem-based learning supports the integration of computational thinking in skill-based mathematics classrooms in an effective way. Table II presents a summary of the key theoretical bases supporting this integration.

TABLE II. THEORETICAL FRAMEWORKS FOR CT INTEGRATION

Theoretical Frameworks	Key Features	How it Embeds CT	Benefits of Skill-Based Learning
Constructivism	Individuals learn and construct knowledge through their own understanding. Here active, and student-centered learning is more emphasized	Students are involved in hand-on exploration. It enables students to actively examine, test and deal with computational problems. It also involves coding, debugging, developing algorithm and refinement of solutions.	It promotes participation, interaction, communication and this helps to develop creativity, critical thinking, and collaboration in students.
Problem-Based Learning (PBL)	Students are assigned an open-ended problem with ill-defined task and work collectively to arrive at meaningful outcomes.	It helps learners to break large, critical problems into smaller components (decomposition) and then create a systemized, step-by-step solutions for the problem (algorithmic thinking).	Fosters collaboration, ability to think critically and self-reflection.
Project-Based Learning (PjBL)	In this, students work on a project for a given period of time to solve the problems or questions based on real world scenario	It offers a real-world context to students for the application of CT components such as decomposition, abstraction, and algorithmic thinking to achieve the final goal.	It promotes creativity, collaboration and self-direction, along with considering social and ethical challenges.
Collaborative Learning:	Learners share their ideas, discuss different strategies and coordinate with each other to solve problems.	Brainstorming about the ideas and testing of strategies in small groups for solving problems. Students coordinate for data organization and provide relevant feedback to achieve solution for the computational problems.	Develops a comprehensive understanding, and versatile problem-solving skills in students. It also fosters communication and collaboration.

1) Constructivism:

According to this learning theory instead of passively receiving information, students actively construct knowledge and understanding through their own experiences and new knowledge is created through previously acquired knowledge and along with this, it promotes active and student-centered learning (Nino, 2023; Olusegun, 2015). It helps learners to enhance their competence or expertise, going beyond the superficial memorization, and to build the basis for interpreting the content (Angraini et al., 2024). It also promotes social interaction and collaboration, which provides a learning environment for the students to share, to discuss, and work together to solve problems. The above characteristics of constructivism boost the development of CT skills in students. Development of practical computational skills depends upon the hands-on experience. Besides this constructivism also fosters critical thinking, which is crucial for understanding the underlying principle of computing. Constructivism promotes critical thinking by motivating students to ask questions, analyze situations, and formulate their own learning experiences. In constructivism, teachers are the guides or facilitators, instead of simply disseminating knowledge their role is to guide students to explore more and solve problem (Angraini et al., 2024). The focus of constructivism on active knowledge construction, critical thinking, and social interaction (Nino, 2023) aligns directly with the hands-on experience and iterative nature of computational thinking (CT). Debugging and iterative refinement skills of computational thinking (Angraini et al., 2024) at its core are

constructivist approaches, as students build and refine their understanding through active engagement and immediate feedback.

2) *Problem-Based Learning (PBL):*

The Problem-Based Learning (PBL) is the learning approach that supports students to inculcate a comprehensive understanding of the concept which in turn fosters critical thinking in students. It engages students with real-world problems rather than abstract tasks (Cronin, 2025). In this approach students analyses concepts deeply and try to find various possible solutions of the problem and then reflect on their outcomes (Staff, 2022). Computational Thinking (CT) is a process problem solving and Problem based learning provides a favourable environment for applying CT skills such as decomposition, pattern recognition, and algorithmic thinking (Barcelos et al., 2018) which results in not only fostering the engagement but also enhancing collaboration or teamwork, communication or effective interaction, and real-world application of knowledge (Team, 2024). CT processes like decomposition, abstraction, and algorithmic design are involved in the complex, real world challenges, so the students develop these skills naturally by working of complex problems (Cronin, 2025; Sheldon, 2017). Students develop not only cognitive skills but also necessary soft skills by applying CT principles directly to real problems (Suparman et al., 2025). This enhances both motivation and learning outcomes of students by meaningful application of mathematical concepts rather than focusing only on theory intensive exercises (Team, 2024). Researches show that PBL provides robust support for the growth of CT skills. In a meta-analysis it was found that both digital (plugged) and non-digital (unplugged) PBL activities substantially enhance problem-solving, where it was found that digital activities have more substantial impact.

3) *Project-Based Learning:*

Project-based learning is closely linked with the Problem-based learning, and is based on the Papert's theory of Constructionism, which shows that when students create something concrete and easy to share, they grasp concepts most effectively. In this approach learners not only solve the problem but also which forms the basis of their learning experience. This approach is applied by the Bootstrap program, where students design a video game using algebra and turn the abstract ideas into tangible and significant outcomes (Navarro & de Sousa, 2023).

4) *Collaborative Learning:*

Designing CT integrated activities provides an environment that fosters collaborative learning, where learners share their ideas, discuss different strategies and coordinate with each other to solve problems. In this process not only the problem-solving skills of students is enhanced but also it develops interpersonal communication and collaboration among students. These abilities are in high demand in the professional settings of the modern world. Most of the unplugged activities are designed based on the collaborative learning approach. For example, games like TrueBall enhances logical thinking and communication skills and games like ShapeGame foster teamwork and self-reflection (Cummins, 2023; European Union, 2025). Similarly, 3D models which are also an example of collaborative projects empower students to actively apply CT concepts with the help of specific software while working in a group.

D. Synergistic Integration: Interrelationship Between Computational Thinking and Skill-Based Learning in Mathematics Classroom

Computational thinking (CT) develops universal problem-solving abilities in students that explicitly support skill-based learning (SBL), and skill-based learning provides a real-world context *for the application of CT principles*, which justifies that CT provides fundamental ground for skill development and skill-based learning is the application of CT in the real-world scenario. CT boosts higher-order thinking skills, which makes the acquisition of practical hands-on skills more systematic. Along with this SBL provides real world context, that makes learning of CT principles or skills more relevant and meaningful.

The core components of CT which are decomposition, pattern recognition, abstraction, and algorithmic thinking (Williams, 2024) align with the principles of Skills-Based Learning (SBL), such as hands-on practice, problem-solving, active participation, and real-world application (Bhasme, 2024). CT is all about solving problem, while SBL is all about acquiring skills and applying these skills and these characteristics of SBL provides suitable environment for CT to flourish (Raj, 2024; Cronin, 2025; Sheldon, 2017).

The modern era of globalization, rapid advancement in the technology, and demand for shifting workplace requires an education system that must develop practical and job-ready skills in students along with theory learning. Skill-based learning straightforwardly fulfills this need by embedding learning in real world applications and also it prepares students for the modern-day challenges of the workplace (Bhasme, 2024).

The non-traditional pathways of skill learning are gaining more importance due to increased cost of higher education and learning of various skills. Now due current demand for skilled workforce many educational institutes are focusing on SBL to prepare graduate students for job market (Williams, 2024). Not only the technical skills but the skills such as communication, collaboration, teamwork, ability to adapt are also required for the job and SBL provides a strong framework such as project-based learning for the development of these soft skills along with the required technical skills (Raj, 2024).

E. *Pedagogical Approaches and Practical Implementation In Mathematics Classroom*

Integration of Computational Thinking (CT) effectively into skill-based mathematics classroom requires coordinated set of approaches for teaching that include various learning styles. There are mainly two strategies for integrating CT into mathematics classroom: 1) Plugged activities, 2) Unplugged activities. These strategies are often integrated within the problem-based or project-based learning approaches. Plugged activities are those that use digital tools such as computer programming languages, educational robotics or virtual labs to engage students for application of CT skills in participatory, engaging and tech-enabled environments. Unplugged activities are those that do not use computers or digital technology, these are the screen free tasks, such as games, puzzles, and manipulatives. These activities support students to grasp CT concepts in a real, observable way.

1) *Plugged Activities: Leveraging Technology (e.g., Scratch, Simulations, Data Analysis)*

Computers and various other digital tools are used in the plugged activities to establish a deeper understanding of the mathematical concepts for meaningful leaning (Mohmad & Maat, 2024). Here, technology is not only the prerequisite, it acts as a booster for learning CT in mathematics.

a) *Programming Environments:* For the introduction of CT skills in math Block-based languages such as Scratch is mostly used (Nordby et al., 2024). In game-based, collaborative, and project-based learning, students are engaged in creating programs to solve problems (Mohmad & Maat, 2024). For example, MATLAB could be used by the higher education students for calculating geometric properties and scratch could be used by the elementary students to model mathematical relationships dynamically (Gadanidis et al., 2017).

b) *Simulations:* Simulations can help learners to explore abstract mathematical ideas. Computer simulations including simulations with Augmented Reality makes the learning more meaningful by the manipulation of variables and observation of outcomes (Mohmad & Maat, 2024). For examples, use of pendulum simulations to study mathematical relationships, also modeling of population growth or decay, where learners gather and analyze data to recognize recurring patterns (Waterman et al., n.d.).

c) *Data Analysis and Visualization:* With the help of software tools students can works with real-world datasets and can learn the statistical concepts like mean, median, mode, variance and standard deviation (Mohmad & Maat, 2024). Software tools can make charts and graphs, which help learners to visualize and communicate findings in an effective way, and these tools help to link the theory of mathematics in practical situations (Team, 2022).

Plugged activities should not overshadow the underlying concepts while allowing complex problem-solving, immediate feedback, and visual exploration and it further leads to the expansion of CT applications. In Plugged activities the technology should systematically support the learning, not dominate it (Liu, 2023).

2) *Unplugged Activities for Foundational CT Skills*

Unplugged activities are those activities that do not require computers or digital tools (Kotsopoulos et al., 2017). These are the important activities because they help to reduce the common challenges in learning CT like necessity to learn a programming language or technological access, and it makes the CT accessible for young learners or beginners (Mohmad & Maat, 2024). These are the hands-on activities which involve the physical engagement of students with the abstract concepts of mathematics, it promotes the collaborative tasks or teamwork while learning (Kotsopoulos et al., 2017).

3) *Examples of unplugged CT activities in mathematics include:*

a) *Algorithmic thinking:* In algorithmic thinking step-by step instructions are followed by the students. For example, an exercise of “random walk” can be designed in which dice can be used to determine direction and steps, which in turn helps students to understand the structure of algorithms (Mohmad & Maat, 2024).

b) *Game-based approaches:* Various game-based approaches are used in for developing the CT skills, for example game like creating a cipher wheel to teach skills like algorithmic thinking (Team, 2022).

c) *Concrete teaching aids:* Various physical teaching aids and activities are used to develop CT skills like decomposition or pattern recognition (Mohmad & Maat, 2024).

The unplugged activities that enhance CT shows that computational thinking is not only about coding or programming, it is about inculcating the thinking process. Through unplugged activities fundamental CT skills can be introduced at an initial level or early stages of schooling, providing students the cognitive skills to handle more difficult tasks at the later stage digitally. It helps to overcome certain technological barriers of CT and allows less digitally equipped schools and less technologically expert teachers to introduce CT concepts in an effective way (Arslantaş, 2024; Kotsopoulos et al., 2017).

After the review of a wide range of literature related to CT it is found that a lot of researches are based on sophisticated, technology-based interventions like robotics and block-based programming (Acar & Dikkartin Övez, 2022; Büscher, 2025; Çelik & Sevgi, 2025; Harlow et al., 2015; Pan et al., 2024; Piedade & Dorotea, 2023; Rodríguez-Martínez et al., 2020; Subramaniam et al., 2022), with fewer studies exploring simple, low-tech approaches. Review of the literature reveals that unplugged activities also fosters creative problem-solving and enriches mathematical understanding (Aytekin & Topcu, 2024). A lot of the studies related to computational thinking have been done outside India with only one project (Iyer, 2019) in India which focuses on the computer science, which highlights a knowledge gap in India. Table III presents the pedagogical strategies for integration of computational thinking in mathematics.

TABLE III. PEDAGOGICAL STRATEGIES FOR INTEGRATING CT IN MATHEMATICS

Strategy Type	Key Features	Strengths	Weaknesses	Benefits for CT & Math
Programming	It includes designing game-based, collaborative and project-based learning by the use of programming languages such as Scratch, Python, MATLAB) to solve the mathematical problems.	It fosters skill-based learning and helps to acquire various skills, offers instantaneous feedback, and enables personalized learning and reflective thinking	It is completely technology dependent and requires tech-competent teachers, also is more focused on the use of digital tools like computers.	Enhances the understanding of mathematical concepts and deepens the learning experiences. Makes students familiar with the programming languages and computational practices
Simulations	Using computer-based simulations, like AR explain the concepts and experiment with variables to monitor effects.			Facilitates the understanding of abstract math concepts by offering hands-on and real-world experiences. Supports examining the relationships between mathematical ideas, patterns and functions.
Data Analysis	Use of various software to learn statistical concepts and analyze the given data to identify the trends in datasets and generating visual representation.			Integrates the mathematical concepts with real-world situations. Cultivates statistical understanding and data interpretation skills in students
Unplugged Activities	Activities that do not involve the use of computers, fundamental and convenient to use. These activities are collaborative in nature and involves physical activities. It also helps to remove technological barriers.	Requires minimum resources for its implementation User friendly and convenient to use. It promotes fundamental concepts, reasoning ability and collaboration among students.	Feedback is not instant. It is not able to promote all the core components of CT as compared to plugged CT activities.	Helps to develop fundamental CT concepts without the requirement of knowledge of programming languages. Removes the technological barriers and builds the essential cognitive skills.

F. Challenges and Considerations for Effective Implementation

Integration of Computational thinking (CT) into skill-based mathematics classrooms leads to several benefits for students, but along with this there are various challenges that needs to be addressed while implementing CT to make its implementation successful. The major challenges are teacher competence, design of curriculum, technological barriers, student engagement, and assessment strategies.

1) *Teacher Readiness and Professional Development*

One of the most important challenge in the CT implementation in classrooms is the lack of teachers' knowledge about CT. Most of the teachers have basic understanding of CT and they often confuse it with the other concepts (Mohmad & Maat, 2024; Saidin et al., 2021). Even the designed curricula can fail, without strong pedagogical content knowledge in CT (Fatimah et al., 2025; Liu, 2023). This barrier can be only removed by giving proper training to teachers about the concepts of CT and ways to integrate it effectively in mathematics classroom. Effective professional development of teachers should be content-oriented, participatory, engaging, cooperative and team-based (Liu, 2023; Yadav et al., 2017).

2) *Curriculum Design and Adaptation*

The curriculum of CT integration into mathematics should be planned carefully which confirm continuous and recurring opportunities for problem-solving across different grade levels (Sheldon, 2017). Rather than integrating CT tools within meaningful mathematical activities, its use in isolation is also a common challenge. Review of the literatures demonstrates that many researches focus on designing CT activities that focus on geometry, and the other domains are left unresearched (Fatimah et al., 2025; Nordby et al., 2024).

Curriculum should be designed in such a way that it should link CT components effortlessly to mathematical practices. It should enable students to understand the reason ("why") behind the implementation ("how") (Raj, 2024). The actual integration of CT in mathematics lessons takes place when it is a part of solving mathematical problems. True integration means designing lessons where CT is part of solving math problems, not an external attachment (Dahshan & Galanti, 2024). Frameworks like PISA 2022 recommends the necessity for designing curriculum that should integrate CT across different mathematical domains, not only geometry and it also focus on blending CT with mathematical thinking (Suters & Suters, 2020; Rahmawati et al., 2024).

3) *Technological Barriers and Infrastructure*

Technological and infrastructural barriers are also a foundational challenge in CT integration. Inadequate availability of computers, lack of internet connections, limited awareness of programming tools can obstruct implementation of CT (Kotsopoulos et al., 2017; Fatimah et al., 2025). Technology is necessary for learning CT but too much stress on digital tools may lead to the deprioritization of CT concepts (Anwar et al., 2024). Combination of substantial and robust pedagogical strategies with a good infrastructure is necessary for effective implementation, recognizing that unplugged activities are vital for cultivating fundamental CT understanding (Liu, 2023; Kotsopoulos et al., 2017).

4) *Student Motivation and Engagement*

There are evidences from the various studies that CT fosters students' engagement, but there are many studies showing that many learners still lack required competence and struggle for acquiring proficiency. Students have superficial understanding of the CT concepts due to which they show less interest and are less motivated (Falk, 2022; Suparman et al., 2025). To overcome these challenges, appropriate teaching strategies should be adopted which should promote discussion, self-assessment and reflective habits in students. With the help of these teaching strategies, teachers can boost students' confidence and increase their motivation for self-regulated learning by enabling them to see how CT contributes to effective problem-solving (Checovich & Licht, 2022; Raj, 2024; Williams, 2024).

5) *Assessment and Evaluation*

Assessment of CT skills is also one of the biggest challenges, old and traditional methods of evaluation like quizzes and exams are not sufficient to assess the CT skills like decomposition, pattern recognition, abstraction or algorithmic thinking. High-quality and impactful assessment of CT skills calls for monitoring problem solving steps, assessing projects and including feedback from peers (Angeli & Giannakos, 2020). Portfolio - based assessment is also a potentially successful approach for CT assessment, it can showcase project samples, student reflections and peer feedback. Rubrics can be designed to assess the different components or aspects of CT such as problem decomposition, pattern recognition, abstraction and algorithmic proficiency (Mariana & Kristanto, 2023; Tang et al., 2020). To develop these assessment frameworks, collaboration between teachers, researchers, and policymakers if required for the high-quality, comprehensive and impactful evaluation of students' growth.

IV. CONCLUSION AND FUTURE DIRECTIONS

Computational thinking (CT) supports learners in acquiring with wide-ranging problem-solving capacity that align closely with skill-based learning (SBL), and skill-based learning (SBL) offers a real-world context for applying

CT, which illustrates that CT provides the fundamental framework for skill development and SBL is the application of CT in real world context. The empirical evidence supporting the benefits of CT integration is strong. Researches show that learning CT improves many cognitive skills, including problem-solving, logical reasoning, critical thinking, and creativity (Cummins, 2023). Along with the cognitive skills, CT integration also enhances noncognitive skills such as student engagement (Checovich & Licht, 2022), motivation, collaboration, confidence, and creativity (Hunsaker, 2022). These benefits help learners to build competence or to enhance skills for handling the constraints of a technology-driven world and to transform their identity from consumers to creators. (Shute et al., 2017).

Review of the literature has reflected that in majority of the studies sophisticated pedagogical interventions have been used, involving the use of plugged activities (Acar & Dikkartin Övez, 2022; Büscher, 2025; Çelik & Sevgi, 2025; Harlow et al., 2015; Pan et al., 2024; Piedade & Dorotea, 2023; Rodríguez-Martínez et al., 2020; Subramaniam et al., 2022) such as robotics, programming like block-based programming, etc., also a few studies had their focus on accessible and simple interventions without the involvement of technology or an enriched classroom setup. The unplugged activities were also observed to boost the students' creative problem-solving skills, and further strengthening their mathematical understanding (Aytekin & Topcu, 2024). Further, the studies are largely conducted in a foreign context, with very sparse studies in the Indian context (Iyer, 2019), clearly indicating the knowledge gap.

The major reasons behind this gap are the lack of adequate teacher training, curriculum design, resource limitations, and students' attitudes, which are significant challenges in the implementation of CT-integrated teaching and learning, but they can be overcome. By redefining professional development, designing flexible curricula, ensuring fair access to resources, encouraging innovation, and developing real assessment frameworks, schools can create a learning environment where students are not just consumers of technology but also its creative builders. The future of education and development of our students and society is influenced by our aptitude to adopt this new integrated approach of mathematics learning and this can be done by realigning our education system with this integrated approach for supporting holistic, lifelong learning. This process of adoption may be complicated, but the aim to achieve an education system that supports every student to attain their highest potential is within our reach.

REFERENCES

- [1] Acar, İ. G., & Dikkartin Övez, F. (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>
- [2] Angeli, C., & Giannakos, M. (2020). Computational thinking education: Issues and challenges. In *Computers in Human Behavior* (Vol. 105). Elsevier Ltd. <https://doi.org/10.1016/j.chb.2019.106185>
- [3] Angraini, L. M., Kania, N., & Gürbüz, F. (2024). Students' Proficiency in Computational Thinking Through Constructivist Learning Theory. *International Journal of Mathematics and Mathematics Education*, 45–59. <https://doi.org/10.56855/ijmme.v2i1.963>
- [4] Anwar, V. N., Darhim, D., Suhendra, S., & Nurlaelah, E. (2024). Exploring the Characteristics of the Digital Pedagogy Model for Developing Computational Thinking in Mathematical Problem Solving. *JTAM (Jurnal Teori Dan Aplikasi Matematika)*, 8(1), 137. <https://doi.org/10.31764/jtam.v8i1.17419>
- [5] Arslantas, T. K. (2024). Theoretical Framework for Integrating Computational Thinking in Education. *Integrating Computational Thinking Through Design-Based Learning*, 15-31. [10.1007/978-981-96-0853-9_2](https://doi.org/10.1007/978-981-96-0853-9_2)
- [6] Aytekin, A., & Topçu, M. S. (2024). Improving 6th Grade Students' Creative Problem-Solving Skills Through Plugged and Unplugged Computational Thinking Approaches. *Journal of Science Education and Technology*, 33(6), 867–891. <https://doi.org/10.1007/s10956-024-10130-y>
- [7] Barcelos, T. S., Munoz, R., Villarroel, R., Merino, E., & Silveira, I. F. (2018). Mathematics Learning through Computational Thinking Activities: A Systematic Literature Review. In *Journal of Universal Computer Science*, 24(7), 815-845. https://www.jucs.org/jucs_24_7/mathematics_learning_through_computational/jucs_24_07_0815_0845_barcelos.pdf
- [8] Bartolini Bussi, M. G., & Baccaglioni-Frank, A. (2015). Geometry in early years: sowing seeds for a mathematical definition of squares and rectangles. *ZDM Mathematics Education*, 47(3), 391–405. <https://doi.org/10.1007/s11858-014-0636-5>

- [9] Bhasme, C. (2024). Significance of Skill-Based Education. In *International Journal of Current Science (IJCS PUB)*, 14(2), 151-156. www.ijcspub.org
- [10] Brackmann, C. P., Moreno-León, J., Román-González, M., Casali, A., Robles, G., & Barone, D. (2017). Development of computational thinking skills through unplugged activities in primary school. *ACM International Conference Proceeding Series*, 65–72. <https://doi.org/10.1145/3137065.3137069>
- [11] Büscher, C. (2025). Differences in Students' Computational Thinking Activities when Designing an Algorithm for Drawing Plane Figures. *International Journal of Science and Mathematics Education*, 23(2), 365–386. <https://doi.org/10.1007/s10763-024-10465-3>
- [12] Cansu, F. K., & Cansu, S. K. (2019). An Overview of Computational Thinking. *International Journal of Computer Science Education in Schools*, 3(1), 17–30. <https://doi.org/10.21585/ijcses.v3i1.53>
- [13] Çelik, E., & Sevgi, S. (2025). Middle school students' views on mathematical situations designed with Scratch. *Journal of Educational Technology and Online Learning*, 8(2), 190–205. <https://doi.org/10.31681/jetol.1636413>
- [14] Checovich, L., & Licht, I. H. (2022, September 6). *Applying computational thinking to boost student engagement in middle school math*. IES Blog. U.S. Department of Education, Institute of Education Sciences. <https://ies.ed.gov/learn/blog/applying-computational-thinking-boost-student-engagement-middle-school-math>
- [15] Cronin, Z. (2025, February 7). *Unlock Mathematical Potential: The Ultimate Guide to Problem-Based Math Learning*. Kiddom. <https://www.kiddom.co/resources/the-guide-to-problem-based-math-learning>
- [16] Cummins, K. (2023, August 16). *Five Reasons Why Computational Thinking Is An Essential Tool For Teachers And Students*. Innovative Teaching Ideas. <https://innovativeteachingideas.com/blog/five-reasons-why-computational-thinking-is-an-essential-tool-for-teachers-and-students/>
- [17] Dahshan, M., & Galanti, T. (2024). Teachers in the Loop: Integrating Computational Thinking and Mathematics to Build Early Place Value Understanding. *Education Sciences*, 14(2). <https://doi.org/10.3390/educsci14020201>
- [18] Delal, H., & Oner, D. (2020). Developing middle school students' computational thinking skills using unplugged computing activities. *Informatics in Education*, 19(1), 1–13. <https://doi.org/10.15388/INFEDU.2020.01>
- [19] Denning, P. J. (2009). The profession of IT: Beyond computational thinking. *Communications of the ACM*, 52(6), 28–30. <https://doi.org/10.1145/1516046.1516054>
- [20] Dyah Rahmawati, R., Nur Wangid, M., & Wahyu Purnomo, Y. (2024). Designing Model of Mathematics Instruction Based on Computational Thinking and Mathematical Thinking for Elementary School Students. In *Mathematics Teaching Research Journal* 143 Early Spring, 16(1), 143-166. <https://files.commonsworld.org/wordpress/wp-content/uploads/2024/06/7.pdf>
- [21] European Union. (2025, June 23). *Building computational thinking without screens: Unplugged activities for students*. Digital First. <https://digitalfirstnetwork.eu/building-computational-thinking-without-screens-unplugged-activities-for-students/>
- [22] Falk, E. (2022, September 15). *How to Engage Students with Computational Thinking Practices in Math*. Digital Promise. <https://digitalpromise.org/2022/09/15/how-to-engage-students-with-computational-thinking-practices-in-math/>
- [23] Fatimah, S., Aulia, M., Dahlan, J. A., & Wahab, A. (2025). *Exploring Research on Computational Thinking in Mathematics: Trends, Challenges, and Impact on Modern Learning*. 26(2), 1196–1218. <https://doi.org/10.23960/jpmipa/v26i2.pp1196-1218>
- [24] Fauzi, A. L., Kusumah, Y. S., Nurlaelah, E., & Juandi, D. (2024). Computational Thinking in Mathematics Education : A Systematic Literature Review on its Implementation and Impact on Students' Learning. *Jurnal Kependidikan: Jurnal Hasil Penelitian Dan Kajian Kepustakaan Di Bidang Pendidikan, Pengajaran Dan Pembelajaran*, 10(2), 640. <https://doi.org/10.33394/jk.v10i2.11140>
- [25] Ferreira, T. D., Soares, J., Santos, D., de Ará Ujo Medeiros, R., Andrade, W. L., Brunet, J., Ramos, M., & Melo, A. (2023). Exploring the Relationship of Mathematical Reasoning and Computational Thinking. *Congresso Brasileiro De Informatica Na Educacao*. <https://sol.sbc.org.br/index.php/wie/article/download/26343/26166/>

- [26] Fiş Erümit, S. (2024). Collaboration of Unplugged and Plugged Activities for Primary School Students: Developing Computational Thinking Skills with Programming. *International Journal of Computer Science Education in Schools*, 6(3). <https://doi.org/10.21585/ijces.v6i3.173>
- [27] Gadanidis, G., Cendros, R., Floyd, L., & Namukasa, I. (2017). Computational thinking in mathematics teacher education. *Contemporary Issues in Technology and Teacher Education*, 17(4), 458–477. <https://citejournal.org/volume-17/issue-4-17/mathematics/computational-thinking-in-mathematics-teacher-education/>
- [28] Han, J. (2023). *A Systematic Review of Computational Thinking Assessment in the Context of 21st Century Skills*, pp. 271–283. https://doi.org/10.2991/978-2-38476-068-8_34
- [29] Hansen, N. K., & Hadjerrouit, S. (2021). Exploring Students' Computational Thinking For Mathematical Problem-Solving: A Case Study. 18th International Conference on Cognition and Exploratory Learning in Digital Age, 251–260. <https://files.eric.ed.gov/fulltext/ED621547.pdf>
- [30] Harlow, D. B., Dwyer, H., Hansen, A. K., Hill, C., Iveland, A., Leak, A. E., & Franklin, D. M. (2015). Computer programming in elementary and middle school: Connections across content. In *Improving K-12 STEM Education Outcomes through Technological Integration*, pp. 337–361. IGI Global. <https://doi.org/10.4018/978-1-4666-9616-7.ch015>
- [31] Haseski, H. I., Ilic, U., & Tugtekin, U. (2018). Defining a New 21st Century Skill-Computational Thinking: Concepts and Trends. *International Education Studies*, 11(4), 29. <https://doi.org/10.5539/ies.v11n4p29>
- [32] Hu, L., & Wang, H. (2024). Unplugged activities in the elementary school mathematics classroom: The effects on students' computational thinking and mathematical creativity. *Thinking Skills and Creativity*, 54. <https://doi.org/10.1016/j.tsc.2024.101653>
- [33] Hunsaker, E. (2022, September 6). *Computational Thinking*. The K-12 Educational Technology Handbook. https://edtechbooks.org/k12handbook/computational_thinking
- [34] Iyer, S. (2019). Teaching-Learning of Computational Thinking in K-12 Schools in India. In *Computational Thinking Education*, pp. 363–382. Springer Singapore. https://doi.org/10.1007/978-981-13-6528-7_20
- [35] Kaup, C. F., Pedersen, P. L., & Tvedebrink, T. (2023). Integrating computational thinking to enhance students' mathematical understanding. *Journal of Pedagogical Research*, 7(2), 127–142. <https://doi.org/10.33902/JPR.202318531>
- [36] Kaup, C. F., Pedersen, P. L., & Tvedebrink, T. (2023). Integrating computational thinking to enhance students' mathematical understanding. *Journal of Pedagogical Research*, 7(2), 127–142. <https://doi.org/10.33902/JPR.202318531>
- [37] Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Yiu, C. (2017). A Pedagogical Framework for Computational Thinking. *Digital Experiences in Mathematics Education*, 3(2), 154–171. <https://doi.org/10.1007/s40751-017-0031-2>
- [38] Kretchmar, J. (2021). *Seymour Papert and Constructionism*. EBSCO. <https://www.ebsco.com/research-starters/religion-and-philosophy/seymour-papert-and-constructionism>
- [39] Liu, Z. (2023). Integrating computational thinking into K-12 education: Bridging the gap between theories and practices. *STEM Education Review*, 1(14), 1–10. <https://doi.org/10.54844/stemer.2023.0467>
- [40] Mariana, E. P., & Kristanto, Y. D. (2023). Integrating STEAM education and computational thinking: Analysis of students' critical and creative thinking skills in an innovative teaching and learning. *Southeast Asia Mathematics Education Journal*, 13(1), 1–18. <https://eric.ed.gov/?id=ED629819>
- [41] Mendrofa, N.K. (2024). Computational Thinking Skills in 21st Century Mathematics Learning. *JJIP - Jurnal Ilmiah Ilmu Pendidikan*, 7(1), 792–801. <https://doi.org/10.54371/jiip.v7i1.3780>
- [42] Ministry of Education. (2020). *National Education Policy 2020*. Government of India. https://www.education.gov.in/sites/upload_files/mhrd/files/NEP_Final_English_0.pdf
- [43] Mohmad, A. F., & Maat, S. M. (2024). Implementation of Computational Thinking Activities in Teaching and Learning of Mathematics in Primary Schools. *International Journal of Academic Research in Business and Social Sciences*, 14(3). <https://doi.org/10.6007/ijarbs.v14-i3/20980>

- [44] Moon, P. F., Himmelsbach, J., Weintrop, D., & Walkoe, J. (2023). Developing preservice teachers' intuitions about computational thinking in a mathematics and science methods course. *Journal of Pedagogical Research*, 7(2), 5–20. <https://doi.org/10.33902/JPR.202318599>
- [45] National Council of Educational Research and Training. (2023). *National Curriculum Framework for School Education 2023*. Ministry of Education, Government of India. https://www.education.gov.in/sites/upload_files/mhrd/files/infocus_slider/NCF-School-Education-Pre-Draft.pdf
- [46] Navarro, E. R., & de Sousa, M. do C. (2023). The concept of computational thinking in mathematics education. *Journal of Mathematics and Science Teacher*, 3(2), em046. <https://doi.org/10.29333/mathsciteacher/13630>
- [47] Nino, T. (2023, March 7–10). *Constructivism as a theory of learning (foundations and significance)*. In *Proceedings of the IX International Scientific and Practical Conference: Basics of learning the latest theories and methods*, 228–232. International Science Group. <https://doi.org/10.46299/ISG.2023.1.9>
- [48] Nordby, S. K., Mifsud, L., & Bjerke, A. H. (2024). Computational thinking in primary mathematics classroom activities. *Frontiers in Education*, 9, 1-14. <https://doi.org/10.3389/educ.2024.1414081>
- [49] Ogegbo, A. A., & Ramnarain, U. (2022). A systematic review of computational thinking in science classrooms. In *Studies in Science Education*, 58(2), pp. 203–230. Routledge. <https://doi.org/10.1080/03057267.2021.1963580>
- [50] Olusegun, S. (2015). *Constructivism Learning Theory: A Paradigm for Teaching and Learning*. 5(6), 66–70. <https://doi.org/10.9790/7388-05616670>
- [51] Pan, Y., Adams, E. L., Ketterlin-Geller, L. R., Larson, E. C., & Clark, C. (2024). Enhancing middle school students' computational thinking competency through game-based learning. *Educational Technology Research and Development*, 72(6), 3391–3419. <https://doi.org/10.1007/s11423-024-10400-x>
- [52] Piedade, J., & Dorotea, N. (2023). Effects of Scratch-Based Activities on 4th-Grade Students' Computational Thinking Skills. *Informatics in Education*, 22(3), 499–523. <https://doi.org/10.15388/infedu.2023.19>
- [53] Rahmawati, R. D., Wangid, M. N., & Purnomo, Y. W. (2024). Designing a Model of Mathematics Instruction Based on Computational Thinking and Mathematical Thinking for Elementary School Students. In *Mathematics Teaching Research Journal* 143 Early Spring, 16(1), 143-166. <https://files.commonsworld.org/wordpress/wp-content/blogs.dir/34462/files/2024/06/7.pdf>
- [54] Raj, S. (2024, December 29). *Skill-Based Education: A Guide to Preparing Students for Real-World Success*. 21K School. <https://www.21kschool.com/us/blog/skill-based-education/>
- [55] Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-López, J. M. (2020). Computational thinking and mathematics using Scratch: an experiment with sixth-grade students. *Interactive Learning Environments*, 28(3), 316–327. <https://doi.org/10.1080/10494820.2019.1612448>
- [56] Saidin, N. D., Khalid, F., Martin, R., Kuppusamy, Y., & Munusamy, N. A. P. (2021). Benefits and challenges of applying computational thinking in education. *International Journal of Information and Education Technology*, 11(5), 248–254. <https://doi.org/10.18178/ijiet.2021.11.5.1519>
- [57] Sheldon, E. (2017, March 30). *Computational Thinking Across the Curriculum*. Edutopia. <https://www.edutopia.org/blog/computational-thinking-across-the-curriculum-eli-sheldon>
- [58] Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. In *Educational Research Review*, 22, 142–158. Elsevier Ltd. <https://doi.org/10.1016/j.edurev.2017.09.003>
- [59] Staff, A. (2022, July 28). *What does Problem-Based Math Learning unlock for Students? Part 1*. Amplify. <https://amplify.com/blog/problem-based-learning/what-does-problem-based-math-learning-unlock-for-students-part-1/>
- [60] Subramaniam, S., Maat, S. M., & Mahmud, M. S. (2022). Computational thinking in mathematics education: A systematic review. In *Cypriot Journal of Educational Sciences*, 17(6), 2029–2044. Birslesik Dunya Yenilik Arastirma ve Yayincilik Merkezi. <https://doi.org/10.18844/cjes.v17i6.7494>
- [61] Suparman, Juandi, D., Turmudi, & Martadiputra, B. A. P. (2025). Plugged versus Unplugged Activities within Problem-based Learning for Computational Thinking Skills: A Meta-Analytic Review. *Edumatsains: Jurnal Pendidikan Matematika Dan Sains*, 10(1), 293–317. <https://doi.org/10.33541/edumats>

- [62] Suparman, S., Juandi, D., Turmudi, T., Martadiputra, B. A. P., Helsa, Y., Masniladevi, M., & Suherman, D. S. (2025). Computational Thinking in Mathematics Instruction Integrated to STEAM Education: A Systematic Review and Meta-Analysis. *TEM Journal*, 14(1), 949–963. <https://doi.org/10.18421/TEM141-84>
- [63] Suters, L., & Suters, H. (2020). Coding for the Core: Computational Thinking and Middle Grades Mathematics. In *Contemporary Issues in Technology and Teacher Education*, 20(3), 435-471. <https://citejournal.org/volume-20/issue-3-20/mathematics/coding-for-the-core-computational-thinking-and-middle-grades-mathematics/>
- [64] Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. *Computers and Education*, 148, 1-22. <https://doi.org/10.1016/j.compedu.2019.103798>
- [65] Team, L. (2022, June 14). *Examples of Computational Thinking in the Classroom*. Learning. <https://www.learning.com/blog/examples-of-computational-thinking-in-the-classroom/>
- [66] Team, L. (2024, August 24). *Defining Computational Thinking*. Learning. <https://www.learning.com/blog/defining-computational-thinking/>
- [67] Waterman, K., Goldsmith, L., Pasquale, M., Goldenberg, E. P., Malyn-Smith, J., Demallie, A., & Lee, I. A. (n.d.). *Integrating Computational Thinking into Elementary Mathematics and Science Curriculum Materials and Instruction*. <https://conference.pixel-online.net/files/foe/ed0008/FP/4648-ICL3076-FP-FOE8.pdf>
- [68] Weintrop, D., Wise Rutstein, D., Bienkowski, M., & McGee, S. (2021). Assessing computational thinking: an overview of the field. In *Computer Science Education*, 31(2), pp. 113–116. Routledge. <https://doi.org/10.1080/08993408.2021.1918380>
- [69] Williams, J. (2024, March 13). *Skills-based learning: Benefits and strategies*. Instructure. <https://www.instructure.com/resources/blog/skills-based-learning-benefits-and-strategie>
- [70] Wing, J. M. (2006). Computational thinking. In *Communications of the ACM*, 49(3), 33–35. Association for Computing Machinery. <https://doi.org/10.1145/1118178.1118215>
- [71] Wu, W. R., & Yang, K. L. (2022). The relationships between computational and mathematical thinking: A review study on tasks. In *Cogent Education*, 9(1). Taylor and Francis Ltd. <https://doi.org/10.1080/2331186X.2022.2098929>
- [72] Yadav, A., Stephenson, C., & Hong, H. (2017, April 1). *Computational Thinking For Teacher Education*. Communications of the ACM. <https://cacm.acm.org/research/computational-thinking-for-teacher-education/>
- [73] Yilmaz, T., & İzmirli, S. (2023). Effect of unplugged and plugged coding activities on secondary school students' computational thinking skills. *Journal of Educational Technology and Online Learning*, 6(4), 1180–1193. <https://doi.org/10.31681/jetol.1375335>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (<http://creativecommons.org/licenses/by-nc/4.0/>), which permits any noncommercial use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

