Computational Thinking and STEM Education

Ebru Kükey

Firat University, Türkiye

Chapter Highlights

This section summary provides the reader with a concise, yet comprehensive overview of the main concepts, frameworks, and findings discussed in the section on computational thinking and STEM education, highlighting its significance, key components, and implications for teaching, learning, and future research.

- 21st Century Skills – The importance of acquiring 21st century skills in today's world
- The Emergence and Importance of Computational Thinking – The importance and emergence of computational thinking in acquiring 21st century skills
- Definition and Components of Computational Thinking – Researchers' explanations of computational thinking and the skills of decomposition, abstraction, algorithmic thinking, debugging, generalization, and evaluation
- Computational Thinking and STEM Education – The connection between computational thinking and STEM education
- Discussion and Conclusion – The integration of computational thinking and STEM education into teaching programs

To Cite This Chapter:

Kükey, E. (2025). Computational thinking and STEM education. In M. T. Hebebci (Ed.), *Innovative Practices in STEM Education: Emerging Technologies, Pedagogies and Learning Models* (123-146). ISRES Publishing

Introduction

The concept of STEM, consisting of mathematics, engineering, science, and technology disciplines is said to have emerged within the last twenty years due to its focus on 21st-century skills (CS) and its incorporation of various disciplines (Langdon et al., 2011). Therefore, STEM education, which facilitates the use of technology and supports the development of 21st-CS through its interdisciplinary approach, is described as a teaching approach that combines engineering, mathematics, technology, and science to transform theory into practice (Meyrick, 2011).

One of the fundamental aims of STEM education is to prepare students with the 21st-CS required to contribute to the economy, foster competitiveness, and develop individuals who will benefit their countries (Morrison, 2006; Williams, 2011). Additionally, the aim is to facilitate learning through STEM education and provide a holistic approach through interdisciplinary connections (Smith & Karr-Kidwell, 2000). STEM education, which covers mathematics, engineering, technology and science, has attracted significant global interest thanks to its potential to develop real-world applications and enhance problem solving skills, as well as its interdisciplinary nature (Margot & Kettler, 2019). STEM education is crucial for achieving the United Nations' Sustainable Development Targets and improving educational standards (Jamali et al., 2022).

21st-Century Skills and Education

In the 21st century, referred to as the digital information age, the importance of developing high-level skills beyond academic knowledge is strongly emphasised (González-Pérez & Ramírez-Montoya, 2022). In this particular context, developing students' skills in algorithmic thinking, abstraction, problem solving, evaluation, and analysis during primary, secondary, and high school is crucial for the development of critical skills in university education (González-Pérez & Ramírez-Montoya, 2022). In the 21st-century, students are expected to be able to generate knowledge and apply it to new problem situations, rather than simply acquiring ready-made knowledge (Wagner, 2008). These characteristics expected of students in today's society are referred to as 21st-CS.

Today, 21st-CS are defined as problem solving, collaboration, critical thinking, communication, adaptability, technological literacy, financial literacy and global competencies (Partnership for 21st Century Skills [P21], 2009). Meanwhile, Lai and Viering (2012) outlined 21st-CS as comprising critical thinking, motivation, creativity, collaboration and metacognitive abilities. They presented the P21 framework alongside these skills, which are crucial for all 21st-century students, as shown in Figure 1 (P21, 2009).

Figure 1. 21st-CS within the P21 Framework

STEM disciplines focus on developing students' communication, adaptability, problem solving, scientific thinking, self-regulation, innovation and creativity skills, thereby emphasising 21st-century learning outcomes (Bybee, 2010a). Students are expected to possess these skills and be able to apply their knowledge in different fields (Nargund-Joshi et al., 2013). Consequently, the structure of STEM education meets the objectives of contemporary teaching programmes (National Research Council [NRC], 2014). In line with this, engineering, innovation, problem solving and computational thinking (CT) are seen as being at the forefront of what is expected of students today (Bybee, 2010b).

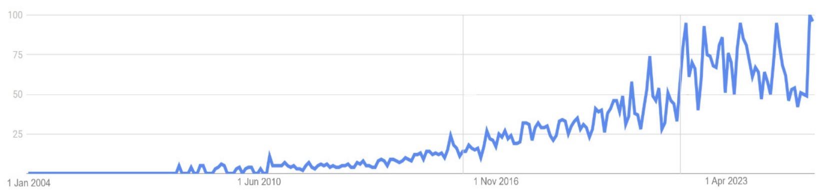

Figure 2. Google Trends search results for "CT" worldwide between 2004 and 2025

CT, which involves problems solving, designing systems and attempting to understand behaviour using fundamental computer science concepts, is one of the 21st-CS (Wing, 2006). The importance of CT is emphasised in the current climate, and research on this concept is increasing. Figure 2 shows a graph obtained by searching for the term "CT" on Google Trends. Examining the figure shows that studies related to CT have gained momentum since

2006, with an increasing research trend in recent years.

The Emergence and Importance of Computational Thinking

It can be stated that the primary researchers responsible for the emergence and significance of CT as a key concept are Seymour Papert and Jeannette Wing (Lodi & Martini, 2021). When CT is considered an ability that students should possess in addition to mathematics, writing, and reading (Wing, 2006), the idea of providing programming education to every student at the K-12 level dates back to Seymour Papert's work at the Massachusetts Institute of Technology in the 1980s.

Papert took the first steps towards what he termed procedural thinking with the aid of programming in his book Mindstorms: Children, 20 Computers, and Powerful Ideas (Papert, 1980). Therefore, it was expressed without a detailed definition by Seymour Papert (1980). Papert conducted research on how to use computer software to solve geometry problems. In this context, Papert (1980) stated that computers develop ways of accessing information and thinking, and contribute to the shaping of learning.

Papert was the first researcher to introduce the concept of CT. Meanwhile, Wing is widely recognised as the first researcher to define the concept using a pedagogical approach, drawing on principles from computer science (Lodi & Martini, 2021). In 2006, Wing provided a clearer definition of CT, stating that it is "the process of using concepts from computer science to understand human behavior, solve problems, and design systems". In this context, Wing (2006) emphasised that CT is a problem solving approach utilising computer science, and that it is a skill set not only for computer scientists, but for all individuals.

Wing (2006) described these skills as iterative thinking, problem solving, decomposition, abstraction, debugging and correction, and reasoning. When describing these characteristics, Wing stated that CT is 'a fundamental ability, not rote learning', 'a conceptualisation, not programming', 'an idea, not artificial', 'thinking like a human, not like a computer', and 'the ability to understand and solve scientific problems through mathematical thinking as an intellectual endeavour'.

CT involves abstraction, debugging and testing, as well as a problem solving approach based on interaction and collaboration with others (Brennan & Resnick, 2012). It is important for education systems to combine students' CT skills with technology to develop their problem solving and thinking abilities (Tsai et al., 2021). In today's technological age, the International Society for Technology in Education (ISTE) has outlined various standards that should be instilled in students based on the need for continuous self-improvement. These standards are provided in Table 1 (ISTE, 2016). Examination of the student standards established by ISTE reveals that students are described as digital citizens, competent learners, knowledge builders, computational thinkers, innovative designers, global collaborators, and creative communicators. Among these characteristics, CT occupies a particularly important position. Computational thinkers are defined as individuals who can use technology to understand and solve problems, develop solutions, and test them.

Table 1. ISTE Student Standards

ISTE STUDENT STANDARDS	
Competent Learner	Students are informed through education and utilize technology to achieve their goals.
Digital Citizen	Students recognize the rights, responsibilities, and opportunities of living, learning, and working in an interconnected digital world and act in safe, legal, and ethical ways.
Constructing knowledge	Students use and critically edit digital tools to construct knowledge, produce creative works, and create meaningful learning experiences for themselves and others.
Innovative design	Students identify problems with new, useful, or creative solutions using technology and create a solution design process.
Computational Thinker	Students use technological methods to understand and solve problems and to develop and test solutions.
Creative Communicator	Students communicate openly and express themselves creatively using platforms, tools, styles, formats, and digital media appropriate to their purposes.
Global Collaborator	Students work effectively in local and global teams, collaborating and broadening their perspectives using digital tools to enrich their learning.

Along with the standards expected to be instilled in students, ISTE (2018) has also defined certain standards for educators. These standards set by ISTE are presented in Table 2. When examining the educator standards, the characteristics of learner, leader, digital citizen, collaborator, designer, facilitator, and analyst come to the fore. It can be stated that educators possessing these characteristics will enable students to develop themselves in terms of the 21st-CS and CT skills they need to acquire.

Table 2. ISTE Educational Standards

ISTE EDUCATIONAL STANDARDS	
Learner	Educators are constantly learning, exploring and implementing proven, effective methods using technology to improve student learning.
Digital Citizen	Educators guide students to develop digital literacy, contribute positively to the digital world, and engage in legal and ethical practices.
Leader	To enhance student competencies, educators must provide technology-enhanced learning. They provide equal opportunities for all students to access digital tools and educational technologies. They demonstrate leadership in the adoption and improvement of new digital educational tools.
Designer	Educators use innovative digital tools and design unique, student-centered activities and environments for active and deep learning.
Collaborator	Educators collaborate with colleagues, parents, and students to develop unique applications using technology, discover and share resources, and solve technology problems.
Facilitator	Educators facilitate learning with technology to support students' competencies according to the ISTE Student Standards, creating opportunities that foster computational thinking in problem solving.
Analyst	By using technology, educators provide students with alternative ways to think about their learning and understand and use data to support them in achieving their learning goals.

When considering the student and educator standards established by ISTE, it is evident that CT processes come to the fore. The characteristics that CT processes should possess can be expressed as follows (Allsop, 2019):

- It is a cognitive process.
- It involves practices that trigger metacognitive thinking.
- It involves the application of knowledge processing thinking skills concepts and abilities.
- Students are expected to apply the concepts they have learned.
- It enables students who are sensitive to problems to develop effective solutions.

What is Computational Thinking?

The definition of CT is debated by many researchers. While some researchers argue that there is no need for a precise definition (Hu, 2011), studies in the field of education indicate that a definition is necessary (Barr & Stephenson, 2011; Hemmendinger, 2010; ISTE, 2011). In this context, Some CT definitions are given in Table 3.

Table 3. Definitions of CT

DEFINITIONS OF COMPUTATIONAL THINKING	
Wing (2006)	It is the process of solving problems, designing systems, and understanding human behavior using fundamental computer science concepts.
Denning (2009)	It is a process that involves thinking at different levels of abstraction, using mathematics to develop algorithms, and examining how well a solution scales across problems of different sizes.
Lu and Fletcher (2009)	It is a conceptual pathway necessary for the systematic, accurate, and efficient processing of information and tasks.
Hemmendinger (2010)	It is to teach them to think like economists, physicists, or artists and to understand how computing can be used to solve problems, create, and discover new questions that can be productively investigated.
Computing Research Association (2010)	It is a rich and evolving set of cognitive skills that are difficult to define precisely.
Wing (2011)	It is a process that requires formulating problems and expressing their solutions by transforming them into information that an information processing tool can effectively apply.
Aho (2012)	It is the implementation of the problem-solving process using computational steps and algorithms.
Furber (2012)	It is the process of recognizing computational aspects of the world around us, using tools and techniques to understand computer science, and being able to reason about both natural and artificial systems and processes.
Kafai and Burke (2013)	Drawing on the fundamental concepts, practices, and perspectives of computer science, computational participation can be expressed as the ability to solve problems with others, design systems with others for others, and understand the cultural and social nature of human behavior.
Yadav et al. (2017)	It is the mental process of abstracting problems and creating automatable solutions.
Manila et al. (2014)	It is the process of formulating and solving problems with the help of a set of computer science concepts and processes.

ISTE (2018) and the Computer Science Teacher Association [CSTA] (2017) have defined CT by highlighting the characteristics in their definitions. In making these definitions, they have also emphasised the five tendencies found in the characteristics of CT (CSTA, 2017; ISTE, 2021). These are:

- Confidence (ability to cope with problems)
- Persistence (ability to solve complex problems)
- Tolerance (ability to cope with uncertainty)
- Ability to cope with open ended problems
- Work ability with a team and communicate in order to achieve a

specific goal

Fundamental Skills in Computational Thinking

Researchers emphasise various skills when considering the characteristics of CT. Figure 3 shows these skills.

Figure 3. Fundemental Skills of CT

These skills are generally accepted to comprise the following components: pattern recognition, decomposition, algorithm creation, abstraction, generalisation and debugging.

Decomposition

Breaking down complex problems into simpler parts for detailed examination and analysis (Wing, 2006). This process can be described as dividing a problem situation into simpler parts (Conery et al., 2011), or building down a problem situation into simpler parts with the intention of reassembling them (Maharani et al., 2019). Csizmadia et al. (2015) describe decomposition as the process of building down a whole into its parts according to specific characteristics, and reflecting on those parts. When dividing a problem into sections, it is important to ensure that the sections can be solved independently and that the solutions can be combined to solve the entire problem (Liskov & Guttag, 2000). Examples of the decomposition step include preparing a simple electrical circuit (Gülbahar et al., 2020) and classifying a species (Barr & Stephenson, 2011).

Pattern Recognition

It is the dimension in which similar or different situations are investigated in the event of a problem (Csizmadia et al., 2015). The identification, classification and analysis of objects according to their characteristics can be described as stages in the pattern discovery process (Liu et al., 2006). In other words, it is the process of dividing problem situations into sub problems and then creating solutions based on the differences and similarities between these sections (British Broadcasting Corporation [BBC], 2019).

Abstraction

Abstraction is described as the foundation of CT and stands out as its distinguishing feature from other forms of thinking (Grover & Pea, 2013). It forms the basis of CT (Wing, 2008). Therefore, it is a dimension in which the problem is simplified by disregarding some of its details (Csizmadia et al., 2015). In this context, Csizmadia and colleagues (2015) explained abstraction ability as the process of making a situation more understandable by reducing unnecessary details.

Abstraction is the ability to decide which details are important and which details will not be included in the process when solving a problem situation (Selby, 2015). Therefore, abstraction becomes critical in determining which details are important when the problem situation becomes more complex (Aho, 2012). Similarly, Wing (2008) also explains abstraction as the decision-making process regarding which details we should emphasise and which details we can ignore throughout the process.

Modelling can be described as an important aspect of abstract thinking. The process of defining variables and modelling equations or inequalities for these variables is an important stage in problem solving and can be considered an abstraction skill (Barr & Stephenson, 2011). Abstraction and decomposition are interrelated processes (Liskov & Guttag, 2000).

Algorithm Creation

Throughout the process, algorithms play a significant role in computer science as they are coherent wholes that produce results, with their operations being clearly, explicitly and effectively sequenced (Schneider et al., 2015). As a method created to solve a problem (Sedgewick & Wayne, 2011), an algorithm refers to a set of logical, sequential operations performed to solve specific problems (Selby & Woollard, 2013). It can also be described as an abstraction of a system involving inputs, a series of steps and goal-oriented outcomes (Wing, 2014).

Once an algorithm has been developed, it can be used to solve similar problems; there is no need to create a new algorithm each time (Csizmadia et al., 2015). For instance, algorithms employed for fundamental mathematical operations can be applied to other problem solving scenarios (Csizmadia et al., 2015). In today's technological age, it has become increasingly important to develop algorithmic thinking among middle school students. This is because, among CT skills, algorithmic thinking enables students to identify appropriate steps for problem situations, find solutions and design solution plans (Neira et al., 2021).

Generalisation

Generalisation is a skill that enables problems to be solved quickly by drawing on previous solutions and constructing solution patterns based on students' prior experiences (Csizmadia et al., 2015). It can be described as a way of finding solutions to new problem situations based on problem situations that have been solved in the past (Csizmadia et al., 2015). Therefore, the skill of generalisation is a strategic skill with applicability, and mastering it is more challenging than mastering other skills (Selby, 2014).

Debugging

Determining whether there are errors in the solutions identified for resolving the problem situation can be described as testing, while identifying and correcting the identified errors can be described as debugging (Maharani et al., 2019). The debugging process constitutes an important dimension of the programming process because it requires a higher level of thinking skill than code writing ability (Liu et al., 2006). In this context, it is crucial to review the process of creating models for solving problem situations, identify errors and correct them when the solutions do not turn out as expected (Angeli et al., 2016). Therefore, CT skills such as debugging are effective in questioning the learning and teaching process and making inferences about problem situations (Gonzales, 2013).

Evaluation

It can be described as the process of determining which of the proposed solutions to problem situations is the best and explaining why the others are not as effective. It also aims to highlight the limitations of the solutions identified (Selby, 2014). Although debugging and evaluation dimensions are expressed as the final dimension among the components, they are dimensions that must be considered in relation to the steps taken at every stage of CT. That is, in problem solving situations, evaluations should be made at every stage and any errors should be debugged (Liu et al., 2006).

Techniques Used in the Application of Computational Thinking

Various techniques can be applied to teach CT skills in the curriculum. By applying these techniques, CT can be transformed into a teaching method separate from computer science. Csizmadia and colleagues (2015) describe these techniques as reflecting, coding, designing, analyzing, and applying (Figure 4).

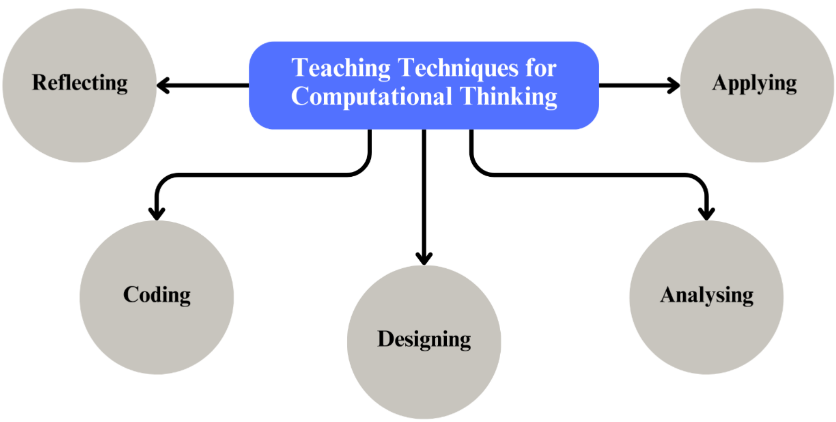

Figure 4. Techniques for Teaching CT

Reflecting

It can be described as the ability to draw honest and fair conclusions in complex problem situations. It establishes general rules and evaluations for determining the most appropriate intuitive methods and criteria for identifying the problem situation within the scope of CT (Csizmadia et al., 2015).

Coding

Coding is a fundamental computer science skill used to evaluate solutions by adapting them to the current problem and achieving a result under the necessary conditions. This allows for a systematic approach involving debugging, predicting outcomes and making logical inferences (Csizmadia et al., 2015).

Designing

Designs created for problem situations aim to determine effective solutions to reach a resolution. These solutions require CT abilities, such as algorithm design, decomposition, and abstraction (Csizmadia et al., 2015).

Analysing

The aim is to find a solution to the problem by breaking it down into subcomponents, eliminating unnecessary details and identifying common and distinct points based on the relationships between components. This process involves making assessments and logical inferences to improve understanding of the situation (Csizmadia et al., 2015).

Applying

It refers to the use of inferences obtained from other problem solving situations to solve a problem. In this way, the characteristics of previously established connections, similarities and differences are utilised. This enables different inferences to be made (Csizmadia et al., 2015).

Computational Thinking and STEM

Although CT originated in computer science, It's a way of looking at things, that has spread to other fields, with specific components and dimensions (Wing, 2008). CT therefore plays a key role in developing skills such as problem solving, collaboration, communication, critical thinking, global competence, technological literacy and financial literacy (Nouri et al., 2020). Developing these skills is said to contribute to individuals becoming competent, confident and determined problem solvers across many disciplines, including mathematics, science and the humanities (Román-González et al., 2018).

CT, recognised as interdisciplinary analytical thinking, is explained as follows: the way of solving a problem is explained by mathematical thinking; the way of designing and evaluating a large and complex system is explained by engineering thinking; and the way of understanding computability, intelligence, and human behaviour is explained by scientific thinking (Wing, 2008). CT enables the modelling of complex structures as well as the analysis of large data sets (Wing, 2008). Based on this, it can be stated that it is a skill that everyone working in many disciplines, such as mathematicians, computer scientists, and engineers, should possess (Wing, 2006).

As emphasised by Wing (2006), CT skills—including abstraction, decomposition, modelling, error correction, logical reasoning, and mathematical and engineering-based thinking—suggest that STEM studies, encompassing science, engineering, mathematics, and technology disciplines, are highly interrelated. The NRC (2010) report lists CT as one of the cognitive characteristics that an individual should possess in modern society, and the 2011 report emphasises that CT is present in all STEM fields and that individuals need to recognise and learn its applications in different fields in order to develop their expertise in this area (NRC, 2011). Furthermore, research has demonstrated the universal applicability of CT across all disciplines (Barr & Stephenson, 2011; Conery et al., 2011; Wing, 2006).

In light of these developments and today's advanced technological structures, it is inevitable that CT concepts and practices will be incorporated into STEM education. Therefore, integrating STEM and CT can greatly enhance people's ability to cope with complex problems (Swaid, 2015). Various studies have examined STEM and CT skills (Jiang et al., 2022; Pewkam & Chamrat, 2022; Shang et al., 2023; Srisangngam & Dechsura, 2020; Qian, 2019). These studies also emphasise that STEM and CT impart skills that facilitate daily life.

According to Thomasian (2011), STEM education has two main purposes.

1. to encourage students graduating from university to pursue careers in STEM fields;
2. to develop the abilities of all students to create creative solutions in daily life using concepts from STEM fields.

STEM education is recognised as an effective way of developing students' problem solving abilities and overcoming the challenge of integrating subjects into the curriculum, while also creating opportunities (Margot & Kettler, 2019). Additionally, STEM education is recognised as an effective means of developing higher-order thinking skills, establishing a professional identity and increasing motivation, while also improving academic achievement

(Li, 2020).

In the 21st century, CT skills have emerged as critical in the development of STEM fields (Law et al., 2021). CT is not merely about processing information; it is a way of thinking that is important in teaching STEM disciplines (Li et al., 2020). Therefore, integrating CT into STEM education has become important for developing students' analytical skills in the problem solving process (Mintii, 2023). It is emphasised that this integration process is also important in educational processes such as STEAM, where STEM education is combined with art (Bell & Bell, 2018).

Discussion

STEM education establishes strong links with 21st-CS and real-life situations (English, 2016). In this context, attention is drawn to 21st-CS in STEM education, and it is argued that 21st-CS should be imparted in all subjects and that STEM education can establish connections across all disciplines (English, 2016). To achieve this, it is necessary to cultivate individuals with 21st-CS such as problem solving, collaboration, creativity and communication, and education systems must be transformed accordingly (Faber et al., 2013; NRC, 2012). STEM education also plays an important role in supporting students' creativity, developing their problem solving abilities, and nurturing productive individuals (Daugherty, 2009).

According to ISTE (2023), CT is a type of literacy that involves problem decomposition, abstraction, pattern recognition and algorithmic properties, and is intended to develop individuals who can find solutions to contemporary problems. Furthermore, it presents the process as a systematic sequence of steps. However, CT is not about thinking like a computer; rather, it focuses on developing all the mental tools necessary for the effective use of computing in solving problems (Lu & Fletcher, 2009). Similarly, CT goes beyond human interaction with and use of computers and technology, encouraging the creation of new explanations, designs for tools, and creativity (Mishra & Yadav, 2013).

Conclusion

Technological developments and modern living conditions are guiding students towards developing themselves in multiple ways. To achieve this, critical skills have come to the forefront. When these abilities are referred to as 21st-CS, CT is actively emphasized by researchers (Aho, 2012; Allsop, 2019; Hu, 2011; Wing, 2006; Wing, 2014).

Wing (2011) defines CT as a thinking process That involves formulating problems and solutions in a way that can be effectively implemented by a computational unit, which is the key to success in this field. The skills expected to be acquired by students have been expressed in various ways based on the definition of CT (Angeli et al., 2016; Barr & Stephenson, 2011; Wing, 2006; Wing, 2011). The following skills come to the fore: differentiation, abstraction, algorithmic thinking, pattern formation, debugging, generalisation and evaluation.

Activities that bring together different disciplines and implement applications are becoming increasingly important in CT development. This situation highlights the importance of STEM education. STEM is an interdisciplinary, student-centred approach to education that uses scientific research methods to solve problems related to daily life (Bender, 2015).

STEM education and interdisciplinary activities are critical for increasing students' interest in various scientific fields and incorporating such education into the curriculum (Knezek et al., 2013). In this context, CT also plays a crucial role in enabling students to establish effective connections. Consequently, it is crucial that teaching programs include content that enables students to develop skills such as algorithmic thinking, decomposition, understanding abstraction, and generalization.

Acknowledgements

This study was prepared solely by the author. No individual, institution, or organization has made any direct contribution to this study.

REFERENCES

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

Allsop, Y. (2019). Assessing computational thinking process using a multiple evaluation approach. *International journal of child-computer interaction, 19*, 30-55. https://doi.org/10.1016/j.ijcci.2018.10.004

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 computational thinking curriculum framework: Implications for teacher knowledge. *Journal of Educational Technology & Society,* 19(3), 47-57.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community? *Acm Inroads*, 2(1), 48-54. https://doi.org/10.1145/1929887.1929905

Bell, J., & Bell, T. (2018). Integrating computational thinking with a music education context. Informatics in Education, 17(2), 151-166. https://doi.org/10.15388/infedu.2018.09

Bender, W. N. (2015). *Differentiating instruction for students*, (Ed.) Thousand Oaks, CA: Corwin Press.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *In Proceedings of the 2012 annual meeting of the American educational research association, Vancouver, Canada.*

British Broadcasting Corporation [BBC]. (2019). Pattern recognition.

Bybee, R. W. (2010a). What is STEM education? Science, 329(5995), 996-996. https://doi.org/10.1126/science.1194998

Bybee, R. W. (2010b). Advancing STEM education: A 2020 vision. *Technology and Engineering Teacher*, *70*(1), 30.

Computer Science Teacher Association [CSTA] (2017). K-12 Computer Science Standards. Association for Computing Machinery, New York, NY, United States.

Conery, L., Stephenson, C., Barr, D., Barr, V., Harrison, J., James, J., & Sykora, C. (2011). Computational thinking: Leadership toolkit (1st ed.). International Society for Technology in Education.

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., and Woollard, J. (2015). Computational thinking-A guide for teachers. Swindon, Computing at School.

Daugherty, M. K. (2013). The prospect of an "A" in STEM education. *Journal of STEM Education: Innovations and Research*, *14*(2), 10-15.

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

English, L. D. (2016). Advancing mathematics education research within a STEM environment. In K. Makar, S. Dole, J. Visnovska, M. Goos, A. Bennison, & K. Fry (Eds.), Research in mathematics education in Australasia 2012–2015 (pp. 353–371). Singapore: Springer. https://doi.org/10.1007/978-981-10-1419-2_17

Faber, M., Unfried, A., Wiebe, E.N., Corn, J. Townsend, L.W., & Collins, T.L. (2013). Student attitudes toward stem: the development of upper elementary school and middle/high school student surveys. In Paper presented at the 120th ASEE Annual Conference ve Exposition, Atlanta, GA: US.

Furber, S. (2012). *Shut down or restart? The way forward for computing in UK schools.* Technical report, The Royal Society, London.

Gonzalez, J. J. (2013). My Journey With Inquiry-Based Learning. *Journal on Excellence in College Teaching*, *24*(2), 33-50.

González-Pérez, L. I., & Ramírez-Montoya, M.S. (2022). Components of Education 4.0 in 21st Century Skills Frameworks: Systematic Review. Sustainability, 14, 1493. https://doi.org/10.3390/su14031493

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

Hemmendinger, D. (2010). A plea for modesty. *Acm Inroads*, 1(2), 4-7. https://doi.org/10.1145/1805724.1805725

Hu, C. (2011). Computational thinking: what it might mean and what we might do about it. *Proceedings of the 16th annual joint conference on Innovation and Technology in Computer Science Education*. Darmstadt, Germany. https://doi.org/10.1145/1999747.1999811

International Society for Technology in Education [ISTE], (2011). *Operational Definition for Computational Thinking.* https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/CompThinkingFlyer.pdf

International Society for Technology in Education [ISTE], (2016). ISTE Standards For Students. https://www.iste.org/standards/for-students

International Society for Technology in Education [ISTE], (2018). ISTE Standarts for Educators. https://www.iste.org/standards/computational-thinking

International Society for Technology in Education [ISTE], (2023). Computational Thinking. https://www.iste.org/areas-of-focus/computationalthinking-in-the-classroom

Jamali, S. M., Ebrahim, N. A., & Jamali, F. (2022). The role of STEM education in improving the quality of education: A bibliometric study. *International Journal of Technology and Design Education*, *33*(3), 819-840. https://doi.org/10.1007/s10798-022-09762-1

Jiang, H., Atuquil Islam, A. Y. M., Gu, X., & Guan, J. (2022). How do thinking styles and STEM attitudes have effects on computational thinking? A structural equation modeling analysis. *J Res Sci Teach. 2023,* 1-29.

Kafai, Y. B., & Burke, Q. (2013). The social turn in K-12 programming: moving from computational thinking to computational participation. [Conference session]. *In Proceeding of the 44th ACM technical symposium on computer science education* (pp. 603-608). https://doi.org/10.1145/2445196.2445373

Knezek, G., Christensen, R., Tyler-Wood, T., & Periathiruvadi, S. (2013). Impact of environmental power monitoring activities on middle school student perceptions of STEM. *Science Education International, 24*(1), 98-123.

Lai, E. R., & Viering, M. (2012). Assessing 21st century skills: Integrating research findings. *Vancouver, B.C.: Pearson.*

Langdon, D., McKittrick, G., Beede, D., Khan, B., & Dom, M. (2011). STEM: Good jobs now and for future. *U.S. Department of Commerce Economics and Statistics Administration*, *3*(11), 2.

Law, K. E., Mageswary, K., & Rozniza, Z. (2021). Computational thinking in STEM education among matriculation science students. Asia Pacific Journal of Educators and Education, 36(1), 177-194. https://doi.org/10.21315/apjee2021.36.1.10

Li, Y. (2020). Six years of development in promoting identity formation of STEM education as a distinct field. International Journal of STEM Education, 7(1), 59. https://doi.org/10.1186/s40594-020-00257-w

Li, Y., Schoenfeld, A. H., diSessa, A. A., Graesser, A. C., Benson, L. C., English, L. D., & Duschl, R. A. (2020). On computational thinking and STEM education. *Journal for STEM Education Research*, *3*(2), 147-166. https://doi.org/10.1007/s41979-020-00044-w

Liskov, B., & Guttag, J. (2000). Program development in JAVA: Abstraction, Specification and Onject Orient Design. Boston: Adison Wesley.

Liu, J., Sun, J., & Wang, S. (2006). Pattern recognition: An overview. IJCSNS International Journal of Computer Science and Network Security, 6(6), 57-61.

Lodi, M., & Martini, S. (2021). Computational thinking, between Papert and Wing. Science & Education, 30(4), 883-908. https://doi.org/10.1007/s11191-021-00202-5

Lu, J. J., & Fletcher, G. H. (2009). Thinking about computational thinking. ACM SIGCSE Bulletin, 41(1), 260-264. https://doi.org/10.1145/1539024.1508959

Maharani, S., Kholid, M. N., Pradana, L. N., & Nusantara, T. (2019). Problem solving in the context of computational thinking. Infinity Journal, 8(2), 109-116. https://doi.org/10.22460/infinity.v8i2.p109-116

Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, & L., Settle, A. (2014). Computational thinking in K-9 education. In: *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference,* ITiCSE-WGR, 1–29. ACM, New York https://doi.org/10.1145/2713609.2713610

Margot, K. C., & Kettler, T. (2019). Teachers' perception of STEM integration and education: a systematic literature review. *International Journal of STEM Education*, *6*(1), 1-16. https://doi.org/10.1186/s40594-018-0151-2

Meyrick, K. M. (2011). How STEM education improves student learning. *Meridian K-12 School Computer Technologies Journal*, *14*(1), 1–5.

Mintii, M. M. (2023). STEM Education and Personnel Training: Systematic Review. Journal of Physics Conference Series, 2611(1), 012025. https://doi.org/10.1088/1742-6596/2611/1/012025

Mishra, P., & Henriksen, D. (2013). A new approach to defining and measuring creativity: Rethinking technology & creativity in the 21st century. *TechTrends*, 57(5), 10. https://doi.org/10.1007/s11528-013-0685-6

Morrison, J. (2006). Attributes of STEM education: The student, the school, the classroom. *TIES (Teaching Institute for Excellence in STEM)*, *20*, 2-7.

Nargund-Joshi, V., Liu, X., Chowdhary, B., Grant, B., & Smith, E. (2013). Understanding meanings of interdisciplinary science inquiry in an era of next generation science standards. In *annual meeting of National Association for Research in Science Teaching, Rio Grande, Puerto Rico*.

National Research Council [NRC], (2010). *Report of a workshop on the scope and nature of computational thinking*. The National Academies Press, Washington, DC.

National Research Council [NRC], (2011). *Report of a Workshop of Pedagogical Aspects of Computational Thinking.* The National Academies Press, Washington, DC.

National Research Council [NRC], (2012). A Framework for K12 Science Education: Practices, Cross Cutting Concepts, and Core Ideas. The National Academies Press, Washington, DC.

National Research Council [NRC], (2014). STEM learning is everywhere: summary of a convocation on building learning systems. *The National Academies Press*, Washington, DC.

Neira, R. H., Garcia-Iruela, M. & Connolly, C. (2021). Developing and accessing computational thinking in secondary education using a TPACK guided scratch visual execution environment. International Journal of Computer Science Education in Schools, 4(4), 3-23. https://doi.org/10.21585/ijcses.v4i4.98

Nouri, J., Zhang, L., Mannila, L., & Norén, E. (2020). Development of computational thinking, digital competence and 21st century skills when learning programming in K-9. *Education Inquiry*, 11(1), 1-17. https://doi.org/10.1080/20004508.2019.1627844

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

Partnership for 21st Century Skills [P21], (2009). P21 framework definitions.

Pewkam, W., & Chamrat, S. (2022). Pre-service teacher training program of stem-based activities in computing science to develop computational thinking. *Informatics in education. 21*(2), 311–329. https://doi.org/10.15388/infedu.2022.09

Qian, Y. (2019). Effectiveness of explicıt guidance and practıces on abstractıon (EGPA): focus on computatıonal tasks ın a stem-ıntegratıve robotıc educatıon program for fıfth grade students. [Doctoral dissertation, University of Georgia]. UCL Discovery.

Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2018). Extending the nomological network of computational thinking with non-cognitive factors. *Computers in Human Behavior*, 80, 441-459. https://doi.org/10.1016/j.chb.2017.09.030

Schneider, G, Gersting, J., & Brinkman, B. (2015). *Invitation to computer science*. Gale, Cengage Learning.

Sedgewick, R., & Wayne, K. (2011). *Algorithms*. Addison-wesley professional.

Selby, C. C. (2014). *How can the teaching of programming be used to enhance computational thinking skills?* (Doctoral dissertation, University of Southampton).

Selby, C. C. (2015). Relationships: computational thinking, pedagogy of programming, and Bloom's Taxonomy. In Proceedings of the Workshop in Primary and Secondary Computing Education (pp. 80-87). ACM. https://doi.org/10.1145/2818314.2818315

Selby, C., & Woollard, J. (2013). Computational thinking: the developing definition, University of Southampton.

Shang, X., Jiang, Z., Chlang, F. K., Zhang, Y., & Zhu, D. (2023). Effects of robotics STEM camps on rural elementary students' selfefficacy and computational thinking. *Education Tech Research Dev. 71,* 1135–1160. https://doi.org/10.1007/s11423-023-10191-7

Smith, J., & Karr-Kidwell, P. J. (2000). The Interdisciplinary Curriculum: A Literary Review and a Manual for Administrators and Teachers.

Srisangngam, P., & Dechsura, C. (2020). STEM education activities development to promote computational thinking's students. 2020 International STEM Education Conference (iSTEM-Ed 2020). Huahin, Thailand. https://doi.org/10.1109/iSTEM-Ed50324.2020.9332734

Swaid, S. I. (2015). Bringing computational thinking to STEM education. *Procedia Manufacturing*, 3, 3657-3662. https://doi.org/10.1016/j.promfg.2015.07.761

Thomasian, J. (2011). Building a science, technology, engineering, and math education agenda: an update of state actions. *NGA Center for Best Practices*. *Washington.*

Tsai, L. T., Chang, C.C., & Cheng, H. T. (2021). Effect of a STEM-oriented course on students' marine science motivation, interest, and achievements. *Journal of Baltic Science Education, 20(1),* 134-145. https://doi.org/10.33225/jbse/21.20.134

Wagner, T. (2008). Rigor redefined. *Educational Leadership*, *66*(2), 20-24.

Williams, J. (2011). STEM education: Proceed with caution. *Design and Technology Education: An International Journal*, *16*(1), 26-35.

Wing, J. (2014). Computational thinking benefits society. *Journal of Computing Sciences in Colleges*, *24*(6), 6–7.

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

Wing, J. M. (2008). Computational thinking and thinking about computing. Philosophical Transactions of the Royal Society A – Mathematical Physical and Engineering Sciences, 366(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Wing, J. M. (2011). Research Notebook: Computational thinking -what and why? *The Link Magazine,* Carnegie Mellon University, Pittsburgh.

Yadav, A., Good, J., Voogt, J., & Fisser, P. (2017). Computational thinking as an emerging competence domain. *Competence-based vocational and professional education: Bridging the worlds of work and education*, 1051-1067. https://doi.org/10.1007/978-3-319-41713-4_49

About the Author

Ebru Kükey works in the Department of Mathematics Education at Fırat University, Faculty of Education, Türkiye. She holds a Bachelor's degree in Elementary Mathematics Teaching, a Master's degree in Mathematics Education, and a Ph.D. in Mathematics Education. She has participated in numerous studies and has published papers and articles at national and international levels. She continues to conduct research in this field.

ekukey@firat.edu.tr
0000-0002-2130-0884