International Journal of Innovation and Research in Educational Sciences
Volume 8, Issue 5, ISSN (Online) : 2349–5219

Conceptual Framework for Twenty-First Century Learning: Developing Computing and Computational Thinking

Neslihan Usta[1*] and Seref Mirasyedioglu[2]
[1]Bartin University, Department of Education, Mathematics Education, Bartin, Turkey.
[2]Baskent University, Department of Education, Mathematics Education, Ankara, Turkey.
[*]Corresponding author email id: neslihanusta74@gmail.com

Date of publication (dd/mm/yyyy): 25/10/2021

***Abstract* – This study aims to focus on the basic components of computational thinking (CT) and to investigate its relationship with competencies/skills in the 21st century and establish new relationships in the context of developing computational thinking in the educational environments. Based on this approach, in the first stage, definitions of computing, computational theory, computational science and engineering (CSE), and mathematical sciences are given, emphasizing the place and importance of CT in the literature, and their relationships between CT are investigated. In the second stage, the difficulties encountered in the development of computational thinking are mentioned and explanations are made about the development of CT. In this context, the approaches proposed for the development of CT through pedagogical and technological pedagogical content knowledge are explained with discussions and examples are given. With this study, the conceptual framework of the development of CT and its relations with 21st-century competencies are established.**

***Keywords* – 21st-Century Competencies, Computational Thinking, Pedagogical Content Knowledge, Digital Competence, Compulsory Education.**

I. Introduction

In the 21st century, when we face the benefits and challenges of globalization and a knowledge-based economy, scientific and technological innovations are becoming increasingly important. Today, where science and technology are developing rapidly, the need for qualified individuals with the skills required by the 21st century is increasing day by day. For this reason, students in high-tech information societies need to develop their skills in the fields of science, technology, engineering, and computational mathematics.

The importance of CT has been increasing in recent years as digital technologies have become an essential component of human activities in the twenty-first century. For example, national security, economy, and public health are significantly affected by developments in digital technologies. This situation causes governments around the world to include computer science education in their priority policies [1]. Although there is no consensus on its definition, CT has basic elements that are widely accepted in various definitions, such as abstractions, model generalizations, systematic processing of information, symbol systems and representations, and algorithmic concepts of control flow [2].

Although it is not a new concept, the concept of CT was first described in education by Wing (2006). Accordingly, CT involves solving problems, designing systems, and understanding human behavior using fundamental concepts for computer science. CT requires a variety of compelling thinking skills, such as abstraction, decomposition, iterative thinking, problem simplification and transformation, error proofing, and intuitive reasoning, which are not limited to software problems but are necessary to solve universal, complex problems. Therefore, CT represents a universal skill set that should be learned by everyone from different disciplines, not just computer scientists [1].Therefore, computational thinking is seen as a skill set that every student and adult should develop. CT is associated with several other 21st century competencies based on problem-solving, critical thinking, productivity, and creativity ([3], [4]).

This study aims to examine the relationship of CT with competencies in the 21st century by focusing on its basic components and to determine the conceptual framework of the development process of CT by investigating its place in curricula and practice in educational environments. For this purpose, in the first stage, the definitions of computation, computational theory, computational science and engineering (CSE), and mathematical sciences are given, emphasizing the place and importance of CT in the literature, and their relationships with CT are investigated. In the second stage, the difficulties encountered in the development of computational thinking are mentioned and explanations are made about the development of CT. In this context, proposed approaches for the development of CT through pedagogical and technological pedagogical content knowledge are explained with discussions and examples are given. Then, researches, based on the given resources, are conducted on the development of CT in compulsory education and its relationship with digital competence. With this study, the conceptual framework of the development of CT and its relations with 21st-century competencies are established. The model regarding the conceptual framework and stages of this study is shown in Figure 1 and Figure 2. (MS: Mathematical Sciences).

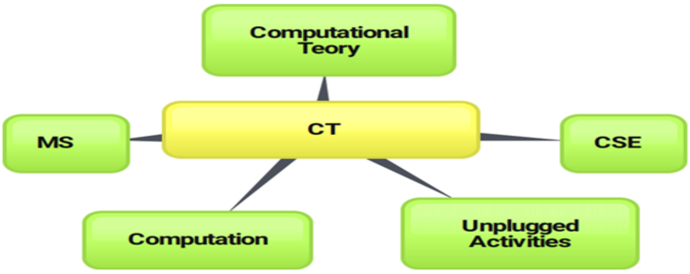

Fig. 1. Disciplinary fields that contributed to the development of ct and their relations with ct.

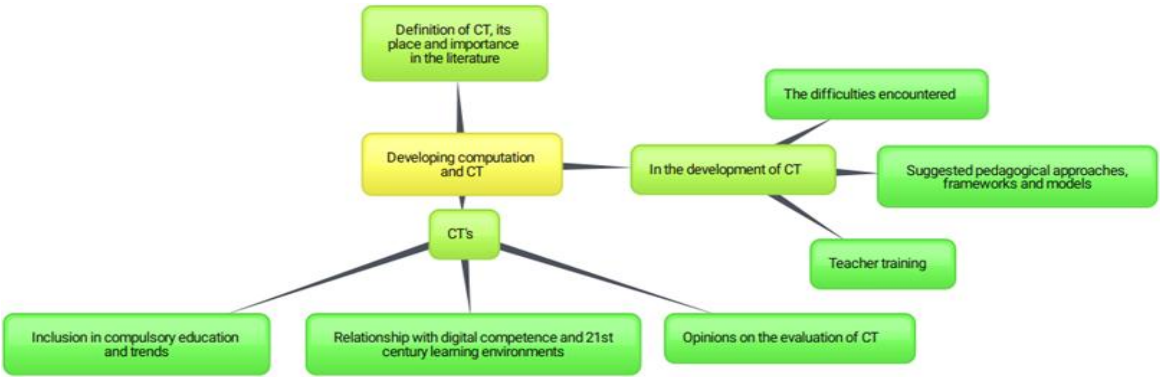

Fig. 2. The conceptual framework and stages of the study.

A. *Theory of Computing and Computational Thinking (CT)*

While making use of concepts that are fundamental to computational thinking, computing, and computer science, it also includes applications such as problem representation, abstraction, decomposition, modeling, testing, and estimation [5]. These applications are at the center of modeling, reasoning, and problem-solving in many scientific and mathematical disciplines [6]. Brennan and Resnick (2012) developed a three-dimensional framework for the analysis of CT. The first dimension is the “computational concepts”, which are the basic elements that people use when programmings, such as order, loops, parallelism, events, conditional concepts, operators, and data. The second dimension contains “computational practice” processes, which include concepts such as abstraction and modularization, reuse, remixing, testing, and debugging. The last dimension is relevant with “computational perspectives”, CT and problem solving, and the views people form about themselves and the environment in which they live. Computing is a process defined by a computational model, and CT is the thought processes involved in formulating problems where solutions can be represented as computational steps and algorithms. Theory of computation, on the other hand, is a branch of computer science that examines whether a problem can be solved with a certain algorithm and computational model, or how quickly and efficiently it will be solved if it is. It has three branches: Complexity Theory, Computability Theory, and Automata Theory [7]. Complexity theory generally deals with decision problems. Decision problems are about being able to answer yes or no to a question. Asking if any number is prime is this type of problem. The reason for examining these types of problems is that any problem can be reduced to a decision problem and thus a solution can be reached. Computability theory examines how much resources are needed to run an algorithm on a problem. This theory investigates how much the running time and memory requirements of the algorithm will increase if the size of the input is increased. The complexity of algorithms is usually measured with time. Automata theory, on the other hand, examines the definitions and properties of different computational models.

According to Maheshwari and Smid (2019), the theory of computing has been and continues to be one of the fundamental areas of applied mathematics. This theory explores the fundamental capabilities and limitations of computational models. A computational model is a mathematical abstraction of a computational system. The most important model of sequential computation studied in computer science is the Turing machine [8] first proposed by Alan Turing (1936). Mathematical abstractions, called computational models, are key to computing and CT. Pascal's and Leibnitz's mechanical calculating machines, Napier's logarithms, Babbage's differentiation, Newman's Colossus, and Turing's crypto analysis for Bombe at Bletchley Park are just a few examples of the computational tools fundamental in the evolution of digital technologies to support mathematical developments [9].

Wing (2006) argues that CT should be added to every child's analytical skill as a vital component of science, technology, engineering, and mathematics (STEM) learning. Dede, Mishra, and Voogt (2013) aimed to advance the discussion on computational thinking by focusing on the basic components of CT, its relationship with 21st-century competencies, and its place in curricula in the EDUsummIT (2013) report. With this report, various professional organizations and think tanks in the USA, UK, and the Netherlands are calling for greater attention to CT in their curriculum. Computation informs politicians in a wide variety of fields such as climate change, public health, and the environment [10].

B. *Computational Science and Engineering (CSE)*

According to CSE Report (2017) and Rude, Willcox, McInnes, and Sterck (2018), computational science and engineering (CSE) is a multidisciplinary field located at the intersection of the core disciplines of mathematics and statistics, science and engineering, and computer science. In this area, the focus is on the integration of the knowledge and methodologies of the disciplinary fields given in the definition of CSE and the development of new ideas at their interfaces. Therefore, CSE is a field itself, different from any of the core disciplines. CSE is a disciplinary field of research into the development and use of computational methods to enable scientific discovery to support innovation in engineering and technology and decision-making in socially important application areas. The role of CSE in the 21st century continues to increase significantly. Going beyond its role in mathematics and physical sciences, CSE drives scientific and technological progress through theory and experimentation and continues its advancement into broad areas including life sciences, medicine, social sciences, business, finance, and government policy ([11], [12]). The combination of CSE's relationship with the basic disciplines of mathematics and statistics, science and engineering, and computer science reveals a new field whose character is different from its original components [12].

According to Rüde et al. (2018), the CSE community is universal and has different aspects. The CSE Core Researchers and Developers group of the CSE community is focused on developing new methods and algorithms such as CSE algorithms, software designs, analysis, development, and testing. This group deals with generally applicable methods or methods developed for a particular field of application. There are two subgroups in the CSE Core Researchers and Developers group. One subgroup focuses on broadly applicable methods and algorithms, and the other focuses on methods and algorithms motivated by a particular field of application. The reason for this distinction is to enable to reveal of differences in desired outcomes for different types of CSE training programs. The CSE Domain Scientists and Engineers group concentrates their work primarily in their science or engineering fields and makes extensive use of CSE methods in research or development. It is widely recognized that CSE will play a critical role in the future of scientific discovery and engineering design.

C. *Mathematical Sciences (MS)*

Mathematical sciences aim to understand the world by reasoning and calculating abstract structures. Mathematical sciences reveal the deep relationships between these abstract structures and make them understood, so they make inferences about the world from the data by using abstract structures and abstract reasoning [13]. At the same time, mathematical sciences act as a natural channel through which concepts, tools, and applications can be carried from field to field. One aspect of the mathematical sciences is to reconnect with related fields in an iterative process to construct abstract structures through the modeling process and make predictions about the world by reasoning on these structures [13].

Mathematical sciences construct a rich and complex ecosystem in which any field contributes to another field and a solution to a problem in one field can emerge from ideas produced in another field. For this reason, the expanding links among many fields of science, engineering, medicine, and business make the mathematical sciences more important. Mathematical sciences are becoming an increasingly integral part and essential component of a growing range of research areas in many fields [13]. These studies involve the broadest integration of mathematics, statistics, and computation and the interaction of these fields with potential applications. All these researches are crucial to economic growth, national competitiveness, and national security, and this fact raises the importance of mathematical sciences in informing both the nature and scale of finance [13].

Researchers are driven to understand the world and find its underlying order and structure. This leads to concepts with deep connections. Some of these connections develop naturally, as much of science and engineering is now based on computation and simulation, where the mathematical sciences are natural languages ([13], [14]). The simulation-based science and engineering process is mathematical and requires advances in the mathematical structures that make modeling possible. Such mathematical structures help in algorithm development, computational fundamental problems, model validation, uncertainty measurement, analysis, and optimization. For scientists and engineers interested in computation and computational theory, advances in these areas are crucial. The ubiquity of computational simulations built on concepts and tools in the mathematical sciences and the exponential increases in the amount of data available to many businesses have become two key drivers of the mathematical sciences [13]. The internet, which has made this huge amount of data available, has magnified the effect of these driving forces. Many areas of science, engineering, and industry are today concerned with constructing and evaluating mathematical models, exploring them computationally, and analyzing a large number of observed and calculated data. All of these researches are mathematical. It is stated that the boundaries in mathematical sciences have begun to be lifted due to the expansion of mathematical sciences and the transition of ideas between sub-fields. As their fields of research become deeply mathematical, many researchers in the natural sciences, social sciences, life sciences, computer science, and engineering are forced to work both in their field and in the mathematical sciences. Mathematical science provides a context for communication and exploration in many other disciplines. Accordingly, the solutions to new problems encountered in technology and various fields of science cannot be solved without associating them with mathematics [14].

Mathematical sciences have an important feature that overlaps with many other disciplines such as science, engineering, and medicine, and increasingly with business fields such as finance and marketing ([13], [14]). In cases where mathematical sciences and their interfaces overlap, there is a mutual interpenetration of research and researchers where two or more disciplines can be combined. This situation necessitates researchers who are connected with mathematical sciences to interact extensively and have a deep understanding of one or more of the overlapping disciplines. Much of modern science and engineering is "mathematical" to a significant degree, with no dividing lines separating the interface fields of the mathematical sciences. For example, people with doctorates in mathematics such as Herbert Hauptman, John Pople, John Nash, and Walter Gilbert have won Nobel prizes in chemistry or economics, and this becomes more common as the field available for mathematical representations increases. With the advancement of mathematical sciences, it is seen that the number of interfaces has increased over time and therefore the mathematical sciences have expanded. In this context, one side of mathematical sciences points to academic science and engineering, while the other side points to social science fields [13].

Mathematics helps to standardize developments in a field for use by scientists in that field [14]. Abstract mathematicians follow natural research paths in the development of new concepts and new theories and are often affected by problems. Concepts and theories developed by abstract mathematicians are needed in many other disciplines in the world of science. For example, integral geometry has led to the development of X-ray tomography; arithmetic structures on prime numbers have led to generating codes for secure data transfer on the Internet; Infinite-dimensional representations of groups have led to the creation of highly connected efficient networks in telecommunications [14].

D. *Computation and Computational Thinking*

In recent years, the spread of computation has transformed the mathematical sciences and application areas. Contrary to theoretical and experimental science, computational science contributes more to science than ever before. Because computational science offers the virtual laboratory where science is applied to the service of science [15]. The computational model synthesizes domain-specific knowledge by using programming structures from computer science, method, and theory from mathematical sciences. Computational science, on the other hand, is used in making and testing engineering designs. This is because computational science allows for high-precision calculations and is predictable. Virtual wind tunnels used in aircraft design and circuit simulation environments used by chip manufacturers are examples [15].

The methods and techniques of computational science have been extended to conceptual models. Computational models have been developed for social and societal systems and other areas that are not generally the subject of mathematical analysis. The ubiquity of computation brings new questions and challenges. Therefore, this situation also transforms the mathematical sciences. Computational science and many applications and basic research involving computation depend on the transformation in mathematical sciences [15].

Associated with fundamental computational models in computer science, there are several well-known algorithm designs and problem-solving techniques that can be used to solve common problems that arise in the data processing. But as computer systems become more complex and apply computer science abstractions to new problem areas, it seems that there are not always suitable models for developing solutions. In this case, CT becomes a research activity that involves inventing suitable new computational models [16]. Priami (2007) states that "the basic feature of computational thinking is an abstraction of reality in such a way that the neglected details in the model make it executable by a machine", which indicates the importance of CT in the creation of computational models.

CT can be defined as "a brain-based activity that enables solving problems, better understanding of situations and better expression of values through systematic abstraction, decomposition, algorithmic design, generalization and evaluation" [17].

International Society for Technology in Education [ISTE] (2015), defines CT as a strong problem-solving approach that combines technology and thought, and a combination of creativity, algorithmic thinking, critical thinking, problem-solving, and collaboration. According to Barcelos Munoz, Villerroel, Merino, and Silveira (2018), CT is a problem-solving approach that represents the problem as an information process according to the computational method and seeks an algorithmic solution.

CT is a new problem-solving method. It uses general computer techniques such as logic, algorithm, discretization, patterns, abstraction, and evaluation. All these techniques help to solve a complex problem [4]. CT allows to stretch the calculation to one's needs and is becoming the new literacy of the 21st century. Therefore, a structured pedagogical framework is required to facilitate effective learning of CT [18].

E. *Principles of Computing and Computational Thinking*

Computing is a broad concept that includes a variety of tasks, concepts, and techniques, while CT is a comprehensive set of approaches and skills with applications in various disciplines [18]. CT shares elements with various other types of thinking, such as algorithmic thinking, engineering thinking, design thinking, and mathematical thinking, and therefore draws on the rich heritage of related frameworks as it expands thinking skills [19]. The dimensions of CT, which are common in different fields of scientific thinking, are classified as abstraction, problem-solving, automation, innovation, and analysis [18]. The principles of computing are computation, communication, coordination, recollection, automation, evaluation, and design [20]. According to Settle and Perkovic (2010), these principles form the basis for recognizing, organizing, categorizing CT samples, and establishing a framework for CT in areas other than computer science. Brennan and Resnick (2012) suggest that computational concepts (arrays, loops, parallelism, events, conditional expressions, operators, and data) are often common in programming interactive media and most common in Scratch programming languages. Stating that computational concepts are not sufficient to explain CT, Brennan, and Resnick (2012) emphasize that computational applications should be made. Computing practices focus on how students learn rather than what they learn. Applications are made by students in this process; incremental and iterative, testing and debugging, reusing and remixing, abstraction and modularisation. According to Brennan and Resnick (2012), expressing, connecting and questioning skills are important in practice. These skills are defined as a computational perspective and the application is structured as interactive media design with Scratch. Scratch programming helps students better understand themselves and their relationships with others and the technological world. Therefore, students are given opportunities to design or construct what they have learned.

Chiasson (2019) investigated the complex relationship between learning areas and the CT process with methods based on a qualitative approach. In the research carried out based on theoretical concepts, how students apply the CT process in robotic learning was analyzed and the contribution of the learning field and its features to the CT learning process was evaluated. The findings showed that innovative learning spaces incorporating new technologies could potentially contribute to CT. Students managed the learning process by making use of the concepts found in the CT model. Approaches to creating multifunctional, comfortable, diversified, and interesting environments in teaching form the basis of 21st-century learning. This study revealed the need for research to bridge the gap between new school models and the needs of society and the school system [21]. Chiasson (2019) stated that his study can be considered as a small-scale exemplary study on what can be done for the effective implementation of teaching based on CT competencies. In the study, it was suggested that teachers help students perceive and investigate problems through feedback to test students' decisions and solve their problems. In such student-centered learning environments, students have the opportunity to conceptualize, construct, experience, verify and share their knowledge through feedback (Freiman et al, 2016; cited in [18]).

II. Computational Thinking: A Pedagogical Perspective

A. *Framework for Computational Thinking as a Pedagogical Device ve Pedagogical Models*

A pedagogical device is defined as a set of rules or procedures by which knowledge is transformed into classroom conversation, curriculum, and online communication [9]. Bernstein (1990) defined the three main areas of the pedagogical device as “the field of production”, “reconceptualization” and “reproduction”. These fields are hierarchically related, so the reconceptualization of knowledge cannot occur without the original production and reproduction of knowledge [9]. Reconceptualization is defined as a process that removes text, signs, or meaning from its original context to fit it into a different context [22].

Knowledge production is a reconceptualization as a result of CS. Reconceptualization gives students the ability to find easy solutions to problems using computers about CT and computational techniques [18]. Despite the relevance of CT to CS, scientists emphasize the importance of teaching CT in disciplines other than CS at e-

-very stage of teaching [23].

According to Chande (2015), the source goal in creating the conceptual framework of CT as a pedagogical device is to consider the components of CT that can provide a systematic direction for applying CT to different problems and gaining CT skills. The aim of including CT as a pedagogical device is to find ways and methods to facilitate learning in all disciplines. Thus, students can be provided with the skills to solve current and future problems. Making the terminology of CS professionals more widely understandable and inclusive is important in imparting CT knowledge and skills. For this, it is necessary to help teachers find ways to integrate their CT knowledge and skills with their existing knowledge and practices, present ideas and materials in contexts that teachers are comfortable with, and communicate with educators using a common language or terminology in the educational environment [18].

According to Lu and Fletcher (2009), proficiency in basic language arts helps to be successful in basic mathematics and to communicate effectively, and CT proficiency helps to process information and tasks systematically and efficiently. However, there are pedagogical difficulties in learning CT [24]. Deciding whether it is necessary to separate programming and basic computer science teaching seems to be the most important challenge. “What is the role of programming in CT, how much programming is required to be taught for CT proficiency?” types of questions should be answered. Also, before students learn a programming language for the first time, a CT foundation should be established. Accordingly, the teaching of CT should focus on the creation of vocabulary and symbols that can be used to explain computation and abstraction, to provide demonstrations of applications and processes around mental models [24].

Pontelli, Brown, Cook, Sanchez, and Nava (2016) explained the purpose of their project, which is based on the fundamental role of computation in all traditional sciences, as instilling CT principles in traditional scientific disciplines, creating new opportunities for scientific discovery and promoting STEM learning. The project aimed to develop a collaborative model between New Mexico State University [NMSU], K-12 schools, and local district schools and to move research into the wider scientific field. For this, K-12 teachers and NMSU faculty members who are outside the CS discipline and have communication and leadership skills were worked with. Pontelli et al. (2016) stated that they could thus develop new educational modules and activities in the curriculum through an innovative collaboration model, and suggested introducing CT as a device to achieve these results. It has been stated that the project will contribute significantly to the development of critical thinking and problem-solving skills of K-12 students, and to enrich the knowledge and skills of middle school and high school STEM teachers in the long run, and will play a role in eliminating the shortage of skilled computing specialists in the USA.

Malyn-Smith and Lee (2012) defined three programs that develop CT skills among students in secondary and post-secondary programs and how the results of the occupational analysis are used in the evaluation of these programs. The framework for this analysis is designed to guide the development of STEM learning. The results of the study showed that K-12 students, equipped with high-quality and effective STEM learning, developed basic CT skills.

A different pedagogical model proposed for the development of CT based on social constructivism theory includes four pedagogical experiences [25]. These are (1) unplugged, (2) tinkering, (3) making, and (4) remixing, respectively. Unplugged experiences focus on activities that are implemented without using computers. Tinkering experiences involve subdividing given tasks and making changes to existing objects. The focuses of the tinkering phase are application, simulation, and problem-solving. Making experiences involve doing activities used to construct new objects. During the making phase, students have the opportunity to solve problems, make plans, make choices, communicate, reflect and make connections between concepts. This phase usually includes practices such as making, prototyping, and testing. Students reveal their learning potential while making and sharing what they make (Dougherty, 2012, cited in [25]). Remixing experiences refers to the use of objects or components of objects for other objects or other purposes. Objects can be digital, tangible, and conceptual. The remixing experience offers students the opportunity to take responsibility for the use and use of digital technologies, to make critical approaches to sampling, computational thinking, and to participate in discussions about the results (Coltor, 2016, cited in [25]). These opportunities are effective in creating digital citizenship [25].

Tsortanidou, Daradoumis, and Barbera (2019) proposed a pedagogical model that aims to combine creativity with new media literacy skills in CT and low-tech and information-rich learning environments. The model will help students gain creativity, problem-solving, and cooperation skills in primary and secondary school settings. In the model, it is stated that teaching students to think like a computer scientist, economist, physicist or artist can be gained through media arts and CT applications. Imagination is expressed as the interface between these applications in the model. In the model, creative teaching methods, unplugged approaches, and low-tech prototyping methods are used to develop students' creativity, CT, collaboration, and media literacy skills. According to Tsortanidou et al. (2019), the model is connected and realized through three macro and five micro-moments. These were determined as non-technological, sociocultural, creative, multimodal, and media. Micro-moments guide specific methods that develop relevant skills and provide natural ways to apply macro-moments. In the model, imagination and questioning are combined through joint design, which highlights the importance of design thinking. In this process, Dewey's ideas on coordinating joint research and imagination processes are used (Steen, 2013, cited in [26]). Dewey states that collaborative design and a collaborative design thought process combine inquiry from the outer world to the inner world and imagination as a trajectory from the inner world to the outer world. From this point of view, it is understood that collaborative design is a reflective process in which something new is produced. On the other hand, it is stated that CT has an important place in increasing creativity as the computer allows the creation of new forms of expression (The College Board, 2012, cited in [26]). The common denominator between design thinking and CT is problem solving and students should solve problems as designers [27]. Problem-solving in CT can be designed in many different ways, from solving practical problems to solving theoretical or conceptual problems [28]. In the model proposed by Tsortanidou et al. (2019), the design thinking process consists of five stages. These stages are; understanding and observing, synthesizing, Ideating, prototyping, and testing. Since intuitive knowledge generates powerful ideas and includes imagination/creativity, learners internalize the world through imagination and externalize it through questioning [26]. Table I includes the thinking process stages and features of the pedagogical model proposed by Tsortanidou et al. (2019). The dimension of associating the process with CT was interpreted in the context of the conceptual framework of CT and added to Table 1 by the authors.

Table 1. Methodologically defined design thinking process and associating the process with CT.

Stages and Features	Associating the stages with the CT
Understanding and observing: It includes understanding the	Students identify sources of errors in a functional model by breaking

Stages and Features	Associating the stages with the CT
context of the problem and the difficulties, feelings, and thoughts of others.	them down into smaller parts to gain a deeper understanding of the parts. Therefore, this stage is related to the decomposition dimension of CT.
Synthesizing: It includes making the final definition of the problem and generating and sharing different ideas to find the source of the problem correctly.	Students solve the problem by analyzing the source of the problem in items, and it will be easier to identify similar errors. Therefore, this stage is related to the pattern recognition dimension of CT.
Ideating: It involves combining students' ideas for problem-solving and their imaginations with meaningful insights.	Students can find possible explanations and solutions through social interaction and feedback from peers and teacher advice and exclude irrelevant weak explanations and solutions by focusing on important details. Therefore, this stage is related to the abstraction dimension of CT.
Prototyping: It includes the conceptualization of possible sources of problems and creative solutions that are translated into real-world solutions through drawing.	It is related to the algorithm dimension of CT, as students construct and implement certain steps or rules of similar logic in the problem and reach a solution.
Testing: It includes testing the accuracy and viability of the imagined solution. The goal at this stage is to identify the exact solution and develop a theory that explains the solution. Issues such as why the model is not functional, i.e. the sources of the problem, how the solution is found, and why the non-functional model works backward are clarified.	It includes checking the functionality and applicability of the model. Therefore, this stage is related to the evaluation dimension of CT.

(Source: [26], [29]).

According to Phillips (2009), the essence of CT is to combine these resources to think about data and ideas and solve problems. According to Phillips (2009), teachers have some duties to perform during the implementation of CT in the classroom:

- They should encourage their students to move beyond using technological tools in their CT and knowledge construction processes.
- They should encourage their students to activate their thought processes to use abstraction and multiple computer science concepts to construct new knowledge.
- They should be able to ask their students different questions about problem-solving and the use of technology in their classrooms, and they should encourage students to ask a lot of questions and construct strategies to solve them. The type of thinking needed by individuals in society is related to the production processes used to solve problems. Knowing how to apply technology to find and use knowledge in solving problems is the first step to improve the process. “How difficult is the problem? How to solve the problem? Which of the CT strategies can be used?” These are some of the questions that teachers should ask their students. In this context, CT provides opportunities and facilitates new ways to solve problems creatively by emphasizing creating knowledge rather than using knowledge [30].
- They should increase their knowledge of ICT (information communication technologies) and help students learn to use ICT to help them understand and solve problems in different disciplines.
- They should help students learn the basic concepts and information about CS.
- They should use computing terms in everyday activities.

Phillips’ (2009) definition of CT and his views on the applicability of CT in the classroom give an idea about how CT should be used as a pedagogical device. It is also important to know the differences that distinguish CT from other learning strategies for an understanding of CT. These differences provide important clues about the necessity of integration of CT into compulsory education. According to Phillips (2009), these differences are listed as follows:

- It takes students beyond technology literacy.
- It creates problem solvers instead of software technicians.
- It emphasizes constructing knowledge rather than using knowledge.
- It offers endless possibilities for solving problems creatively.
- Develops problem-solving techniques already taught.

An exemplary study supporting Philips' (2009) definition and views on CT is the online experimental study by Hadad, Shamir-Inbal, Blau, and Leykin (2020) investigating pedagogical strategies in coding and robotics education. Hadad et al. (2020) examined the development of students' CT skills and the contribution of online courses to the professional development of teachers in the context of coding and robotics. In the study, an online program was applied to increase students' creativity, cooperation, problem-solving and self-expression skills, motivation, and self-confidence. At the end of the one-year practice, it was concluded that the teachers participating in the study showed a development from the teacher-centered teaching process to the student-centered teaching process in their lessons, the difficulties encountered in teaching CT were significantly reduced, the teachers were able to manage their independent learning and teaching processes and provide more guidance to their students. Teachers encouraged their students to learn individually, and by supporting them in this regard, they were able to use different pedagogical approaches together in the teaching process. In this context, the pedagogical strategies teachers use in the teaching process were creating learning experiences, debugging, collaborative learning, gamification, explanation with examples, peer teaching, and teacher lecture strategies. The teachers who participated in the study stated that such an application provided significant benefits in their professional development. According to Hadad et al. (2020), the creation of learning environments in which the student is active takes responsibility for their learning, and can conduct the independent learning process is effective in the development of CT skills. In this process, it is emphasized that it is important for teachers to guide a way that facilitates and helps students discover and construct knowledge.

According to Tikva and Tambouris (2021), CT creates an ideal environment for the development of 21st-century skills through programming. This interest increases the number of studies, but various difficulties arise in the educational applications of CT. In studies in the literature on CT and CT fields, it is seen following issues have not been sufficiently researched yet [31], "Which tools support which learning strategies in educational applications and development of CT, which learning strategies enable the acquisition of CT, what are the factors affecting the development of CT, and how students' CT levels affect capacity building". For this reason, Tikva and Tambouris (2021) developed a systematic conceptual model by mapping the CT curriculum to overcome these difficulties and improve CT. This model emerged from the synthesis of 101 studies and the identification of CT fields. The CT fields of the model are Knowledge Base, Learning Strategies, Evaluation, Tools, Factors, and Capacity Building. The developed model examines the relationship between CT fields. It was emphasized that the model could help the understanding of CT and form the basis for future research studies.

In addition to the models proposed for the conceptualization of CT as a pedagogical device, non-computer applications for the development of CT are also of great importance. Aranda and Ferguson (2018) stated that the use of "unplugged activities" was effective in the conceptual development of CT. Because CT uses the basic concepts of computer science. These concepts are generally related to the problems that students encounter in their daily lives. Therefore, it was stated that students can learn these concepts with "unplugged (not computer-mediated) activities" without using a computer (Sie & Yan, 2017, as cited by [32]). Sie and Yan (2017) stated that it is not necessary to have a computer for students to learn computer concepts and develop their technology competencies [33]. "Unplugged activities" are activities that teach coding concepts without using a computer. With these activities, students can use paper, pencil, various tools, and their bodies [34]. Unplugged Programming (UP) refers to learning CT and computer science concepts without using computing devices [35]. With unplugged activities, students not only simulate their learning processes, but also find opportunities to explore the basic ideas of computer science and do coding with these activities [36]. Unplugged activities require students to learn CT and how to define a problem, identify important details for problem-solving, and use logical reasoning to create and evaluate a problem-solving process [37].

Unplugged activities were initially carried out with "hands-on activities" developed by CS teachers and students without a computer with a pedagogical approach. The most important reason for CS teachers and students to develop this idea was the thought that computers are learned using programming, but this is not motivating enough by every student, sometimes the computer can be a distraction in learning and this creates an obstacle to learning. Thus, it has been shown that computer concepts can be learned without using a computer [35]. These activities, designed for primary school children, have become widespread in educational environments, where students of all ages can successfully use them formally and informally (Earp, 2016, cited in [35]).

Studies have shown that UP can facilitate the learning of CT concepts ([29], [38]). AlAmer et al. (2015) stated that they used UP in teaching CS concepts and that these concepts started to be used more by the participants at the end of the study. Brackmann et al. (2017) conducted a 5-week practice on an experimental group to develop CT skills using UP with a quasi-experimental method. When the CT skill test scores performed a few years later were analyzed, it was seen that the experimental group had a statistically significant difference in terms of overall effect compared to the control group [29]. Studies with the application of UP activities revealed mostly positive results in learning CT concepts, but some studies did not reveal any remarkable results (Campos, Cavalheiro, Foss, Pernas, Piana, Aguiar, Du Bois, & Reiser, 2014, cited by [29]). Although it has been shown that unplugged activities can provide CT's content knowledge gains, there are some criticisms about UP [35]. One of them is AlAlmer et al.'s (2015) comments that criticize the Unplugged program for focusing more on macro aspects (e.g. binary numbers, designing algorithms) than micro aspects of CS (e.g. local and global variables, conditional statements). Bers (2018), on the other hand, criticized UP, which is framed with CT as a problem-solving process, stating that it does not allow a tool that will enable students to express their creative ideas created by an external creation [35]. In this context, more studies are needed to reveal the reasons for these situations.

Students can use tablets and computers to code in the digital world. However, it is noteworthy that the use of UP programs, which are thought to have potential, together with other ways such as coding, will help to learn

CT [35]. Brackmann et al. (2017) state that as a result of their experimental studies without using a computer and using a computer, the students who participate in the activities without using a computer significantly improve their CT skills compared to the students who do not attend these courses, and that the activities without using a computer are effective in gaining these skills. On the other hand, more research is needed to determine the effects of computer-free activities on the acquisition of CT skills on younger students who will encounter CT for the first time [39]. In addition, according to Hsu, Chang, and Hung (2018), since the cognitive capacities of students differ according to their ages, this situation should be taken into account in determining the method, content, and learning strategies to be used in CT teaching.

B. *Developing Computational thinking in Compulsory Education and Major Trends in CT Integration*

In recent years, CT-related concepts (e.g. coding, programming, algorithmic thinking) have received increasing attention in the field of education and this has led to various initiatives in academia, gray literature, and many public and private institutions [40]. Despite this widespread interest, there are unresolved problems and various challenges in the successful integration of CT into compulsory education [40]. Phillips' (2009) definition of CT and his views on the applicability of CT in the classroom give an idea about how CT should be used as a pedagogical device.

The basic questions that need to be considered for the successful and effective integration of CT education into compulsory education can be listed as follows [40].

1. How can CT be described as a key skill of the 21st century for students?
2. What is the relationship between the basic features of CT and programming/coding in compulsory education?
3. How should teachers be trained to effectively integrate CT into teaching practices?
4. Should CT be considered within a particular subject or as a subject in curricula?
5. What does evaluation mean in CT?
6. What is required to develop CT in compulsory education?

According to JRC (2016), CT is seen as a way for students to develop their problem-solving skills. Webb et al. (2016) argue that CS to be included in the curriculum is an ideal way for students to develop CT, which they can later apply broadly as a problem-solving strategy (cited in [40]). While the Informatics Curriculum in Austria sees the understanding of informatics as a way to achieve problem-solving, the French study group emphasizes that students should be active digital citizens who can lead digital transformation rather than being subject to digital transformation. The Australian curriculum states that it is important for students to learn how to use and develop digital technologies to fully participate in the digital world. In this context, it is thought that the introduction of CT will be effective in filling the gap between the curriculum and the current needs of students and society in general [40].

Countries that develop their studies on basic questions (Austria, Czech Republic, Denmark, Finland, France, Greece, Hungary, Italy, Lithuania, Poland, Portugal, Turkey) aim to develop students' logical thinking skills and problem-solving skills [40]. Lithuania aims to improve students' ability to organize and analyze data, while Finland aims to strengthen students' motivation to improve their math reading skills. The Finnish curriculum supports competencies such as CT, learning to learn, cultural competence, ICT competence, and entrepreneurship. Finland, France, Lithuania, Poland, Portugal, Switzerland, and Turkey especially focus on the development of coding and programming skills in their curriculum. The Czech Republic, Finland, Poland, and the Netherlands see the development of CT skills as a way for students to prepare for life in the digital world. In addition, the inclusion of CT in compulsory education is shown as the necessity for school education to follow social developments. In this context, two trends emerge in the rationale for the inclusion of CT in education. First, it is important to develop CT skills to enable children and young people to think multi-dimensionally, express themselves through the media, solve real-world problems and analyze everyday problems from a different perspective. The second is the promotion of CT to meet the current need in Information and Communication Technologies to accelerate the economic growth of countries and to meet future employment [40].

Most of the articles in the literature mention the general benefits of CT as a thinking skill and the necessity of developing new skills needed by the employment market. JCR (2016) emphasizes the need to develop digital skills for employability in its report. European Commission member states recommend that more investment be made in the acquisition of digital skills in education, including coding and CS. Studies on employability and the acquisition of digital skills are also carried out in countries outside of Europe. For example, the Australian curriculum states the need for students to learn how to use and develop digital technologies to fully participate in the digital world [40].

It is emphasized that the main purpose of promoting CT in Europe and many non-European countries is to develop 21st-century skills in students. Accordingly, many countries in Europe have included or are planning to include CT-related concepts in compulsory education as part of their curricula. Countries are divided into three clusters according to the approaches they adopt to include CT and CS in compulsory education [40]. The first cluster includes countries that have begun to review and revise their curricula in the last few years. In the process, such countries have increased the teaching of CT and related concepts in compulsory education at the national level, regardless of whether it poses a social challenge or is suitable for employment market needs. The program often emphasizes and prioritizes the importance of CT concepts and skills related to improving students' coding and programming abilities. In addition, teacher education programs have been focused on supporting the integration of CT and CS into the curriculum. The second cluster includes countries that have not yet included CT in compulsory education but are preparing to do so soon. The third cluster includes countries that rely on long-standing traditions in CS education in Europe, especially in high schools. The trend among these countries is to expand CS education to all levels of education. CT plays a central role in this trend [40].

C. *Opinions on the Integration of Computational thinking into Compulsory Education Curriculum and Teacher Competency*

Today, digital technologies play an important role in fulfilling basic tasks in daily life. For this reason, individuals need to receive the necessary training, have sufficient knowledge and skills, and develop problem-solving skills to critically understand the technological systems they use ([3], [4]). Wing (2006) states that society needs individuals who know what they can and cannot do with computers, and individuals can become effective authors or creators of computational tools. Wing (2006) defined CT as the process of formulating and solving problems using a computer and stated that CT will help K-12 students learn to think abstractly, algorithmically, and logically, and solve complex and poorly structured problems. Angeli et al. (2016) proposed a CT framework based on a holistic design approach consisting of the dimensions of computational thinking such as abstraction, generalization, decomposition, algorithmic thinking, and debugging. They stated that with this framework, they aimed to introduce young students to CT processes and to become competent in learning more theoretical and practical subjects of CS in the following years. It is stated that the framework is conceptualized to allow teachers the opportunity to adapt and customize it as they see fit for their classrooms and students. Van Merriënboer and Kirschner (2007) define a holistic design approach in the curriculum as "trying to deal with complexity and without overlooking the separate elements and the interconnections between these elements" (as cited in [41]). The holistic design promotes segmentation and fragmentation, the breaking up of a whole into small, distinct, and often discrete parts. On the other hand, the holistic design approach focuses on complex and authentic learning tasks without overlooking the individual elements that make up the complex whole, thus aiming to eliminate segmentation and fragmentation. Therefore, when this approach is applied correctly by the teacher, students can learn CT in the problem-solving process [41]. A curriculum designed around real-life problems should be considered as a way to bring CT to a situation in students' lives and to engage them [41].

Asia Pacific Region countries such as Korea, Taiwan, Hong Kong, and China have started their national curriculum reforms by realizing the importance of CT education in K-12 education due to their strength in the ICT industry [1]. For example, although nationwide CT education in Korea is in its early stages, there is evidence from some research supporting the effectiveness of learning coding skills and related CT skills. CT education in Korea has been conducted in the areas of (a) innovative pedagogical approaches specific to CT, (b) developing evaluation tools to measure students' CT knowledge, skills, and attitudes, (c) expanding computing, training educators, and coding, (d)) training teachers' CT skills and their perception of the effectiveness of coding education. The curriculum for CT education in Taiwan introduced in 2019 aims to (a) develop CT skills, (b) build literacy in using information technologies for collaboration and communication, (c) promote appropriate attitude in using information technologies (Ministry of Education, Taiwan, 2016, cited in [1]). In Hong Kong, CT is mainly developed through coding education. With the rapid technological development, CT is recognized as a fundamental problem-solving skill in Hong Kong society. Results of a study conducted at a local primary school in Hong Kong from 2012 to 2014 showed that CT had a positive effect on teaching mathematics and general subjects. However, it was stated in the same study that the lack of teacher education and practical difficulties on curriculum issues in CT applications in formal education should be considered (Wong et al, 2016, cited in [1]).

It is important for teachers to be prepared for new pedagogical approaches and to understand how ICT and pedagogy relate and interact to facilitate the development of these competencies to equip both themselves and their students with 21st-century competencies [40]. However, truly integrating CT into existing curricula presents significant challenges, especially for teachers ([31], [42]). According to Peng, McNess, Thomas, Wu, Zhang, Li, and Tian (2014), the introduction of new ideas and concepts with the new curriculum has revealed the problem of the lack of competent teachers who can implement this curriculum. With the start of the implementation of these programs, teachers stated that they encountered various difficulties while teaching CT and CS subjects [2] and that they had to develop new learning resources and use new technologies, but they did not have much confidence in this issue (Curzon et al, 2009, cited in [42]). Teachers state that they were not ade-

-quately supported in various subjects, especially in the lack of resources and teaching CT skills [43].

The issue of how to integrate computational thinking into the curriculum has not yet been fully clarified. Moreover, the problem of preparing a curriculum in which the design of a curriculum based on a general CT framework, and the technological pedagogical content knowledge explained in detail necessary for teachers to master this curriculum, are among the challenges that need to be resolved ([31], [41]). On the other hand, the participation of teachers in the curriculum renewal process is important. Because this participation is a tool of professional development (Vitikka, n.d., cited in [40]). This will enable teachers to gradually participate in the renewal process, understand the main ideas and not perceive curriculum change as a top-down process with guidelines and regulations imposing [40].

The fact that teachers have a sufficient infrastructure to implement any curriculum determines the quality of teaching. Shulman (1986) emphasizes that a teacher should be competent in various fields of knowledge such as how he teaches, how he reflects his field knowledge to teaching, the ability to know and use teaching strategies, and student and program knowledge. Shulman (1986) defines the knowledge and abilities that teachers should have as pedagogical content knowledge (PCK). PCK is “the blending of content knowledge and pedagogical knowledge for teaching purposes, which includes organizing, presenting, and adapting specific topics or problems for students of different interests and abilities” [44]. For the last two decades, education researchers have expanded the concept of teachers' pedagogical content knowledge to technological pedagogical content knowledge (TPCK) by adding technology knowledge [41]. TPCK, proposed by Angeli and Valanides (2005, 2009), is conceptualized with the contribution of five different knowledge bases: content knowledge, pedagogical knowledge, learners' knowledge, educational context knowledge, and technology knowledge (as cited by [41]).

Mishra and Koehler (2006) argue that the TPCK framework should successfully integrate technology into the curriculum, and technology and pedagogy should be combined with content creatively and holistically. By focusing on ways of thinking, which they call interdisciplinary skills, Mishra, Koehler, and Henriksen (2011) state that the TPCK framework will expand learning in the 21st century and that such a focus will develop creativity in students.

It is important to know the thoughts of teachers about the development of computational thinking. Because the teacher's point of view, knowledge, and ability determine the quality of teaching. The workshop conducted by Bower et al. (2017) on the development of CT skills investigated whether teachers have pedagogical competence in the development of CT and related issues. To the question “Which technologies can be used to develop students’ computational thinking skills?”, it has been observed that the majority of teachers think that technological devices can support the development of CT, and they answered “computer and digital devices should be used” in this process. Digital devices, systems, and networks encourage computational approaches to solve problems [2]. Before the workshop, while the teachers answered the question of “what prevents you from being confident in developing your students' computational thinking skills?” as "lack of knowledge and inability to understand computational thinking", at the end of the study, they stated that their feelings of inadequacy in teaching concepts decreased, but the lack of resources to support teaching was an important problem for them. In addition, the teachers stated that they wanted to know how to integrate CT into the curriculum in the face of the intensity of the curriculum and how to make the time given for teaching the subject and gaining skills sufficient. Teachers explained the factors that would make their students feel confident in developing their CT skills. Accordingly, the teachers stated that they needed more time to learn, that it would be appropriate to provide more resources and support to access the relevant technologies, and that it would be beneficial to present more examples, activities, and ideas on how CT should be taught. In addition, they stated that their self-confidence could increase by preparing environments that allow them to exchange ideas with their colleagues and to enable them to communicate with experts in the field. After the workshop, it was stated that the teachers developed an awareness of the sub-components of CT such as decomposition, pattern recognition, algorithm design and abstraction, and a more detailed understanding of CT. At the end of the study, it was seen that teachers generally focus on student-centered learning approaches towards the types of pedagogy that they can use to develop CT. While the pre-workshop teachers were not aware of the field-specific technologies that could be used to teach CT, the majority of the post-workshop teachers stated that they could use software such as Scratch Junior, Python, and significantly increased their self-confidence in teaching CT [42]. In summary, the results of the study showed that teachers were able to develop the basic ideas of CT content, pedagogy, and technology in a relatively short time and developed significantly better self-efficacy towards related concepts and practices.

According to Bower et al. (2017), widespread criticism is made against teachers that they cannot adequately respond to a new discipline and teaching demands arising from the changing technological workforce and social needs. However, they stated that this criticism did not coincide with the results of their study and that teachers might be quite capable, and that the main problem was to provide them with well-designed professional learning opportunities and resources. The study conducted by Ketelhut, Mills, Hestness, Cabreral, Planel, and McGinnis (2020) to see the integration of CT into the curriculum and classroom application of primary school teachers for their professional development, and to determine how teachers' views change in the experimental process, can be considered as an example that supports Bower et al. (2017) views. According to Ketelhut et al. (2020), students' acquisition of CT skills at an early age is related to their teachers' completion of their development on this subject. Ketelhut et al. (2020) stated that at the end of the study, teachers developed their knowledge, beliefs, and attitudes about the integration of CT into the lesson in a positive way and that they adapted what they learned to the curriculum in the best way by using it in the design of the lesson. In this context, if the teacher is to provide opportunities for his students to gain CT skills, the teacher needs to be able to adapt CT to the curriculum and know-how to apply it in the classroom, and teachers' professional development should be ensured.

In the study conducted by Kong, Lai, and Sun (2020) on the development of primary school teachers' CT skills, the proposed program focuses on subject knowledge and pedagogy and offers an effective teacher training development model with active practices. Evaluation of CT skills in the experimental study was based on objective tests and self-evaluations. As a result of the application, it was observed that there were improvements in the knowledge of teachers about CT concepts and applications, subject content information, technological and technological pedagogical content knowledge. According to Kong et al. (2020), the effect of effective teacher training programs in the development of CT has been experimentally proven with this study. In this context, it is important to organize teacher training courses or in-service courses to improve teachers' technological and pedagogical content knowledge. Attention getter of the study by Kong et al. (2020) is that the schools where the participating teachers work have an infrastructure in which the relationship between programming and CT can be established better than other schools. In other words, it was the fact that the teachers who participated in the teacher training program prepared for the development of programming and CT skills had a certain level of knowledge in the field of computer science (CS). However, in the process, it was observed that there were significant improvements in the development of CT among teachers who were not knowledgeable in CS. On the other hand, to see whether it is possible to generalize the results of the study, it is emphasized that the study should be conducted in schools that do not have the appropriate infrastructure in the development of CT.

Reichert, Couto Barone, and Kist (2020) analyzed a group of K-12 mathematics teachers' achievements in the CT teaching process and their views on the use of CT in mathematics teaching with a qualitative approach. According to this, it was seen that the teachers had deficiencies in the association between mathematical subjects at the beginning, but upon the completion of the training, it was seen that the teachers developed various methods in mathematics teaching and they had positive views on how to apply, learn and teach CT. In the study, CT applications were made with unplugged activities, presentation of basic robotic concepts with Scratch programming, and block-based programming using ArduBlock. In the study, which examines the perceptions and pedagogical approaches of teachers in the integration of CT into the mathematics lesson, how teachers integrate CT into their classrooms, their contribution to the teaching process, their attitudes towards CT, and possible changes from traditional teaching to constructivist teaching were investigated. It was observed that the teachers who participated in the application generally associated CT with problem-solving and made correct definitions. According to Reichert et al. (2020), CT concepts should be included in education programs and disseminated with the support of the media along with developing technology. With such supportive strategies, in-service training to be organized by educational institutions and the state should contribute to the professional development of teachers. Because teachers think that they are supported by such practices and they have confidence in themselves [45].

Gleasman and Kim (2020) state that block-based programming is applicable in teaching mathematical concepts, positive results are obtained in the development of pre-service teachers, and it is important to include a structure consisting of CT, programming, and mathematics teaching in teacher training curricula and to see the results with the applications made. According to Finsterbach Kaup (2020), the technology developing with the digital age has revealed the idea of using digital manipulative tools such as robots in mathematics education. Since digital manipulatives include computational skills, their use in problem-solving approaches helps students develop their CT skills. Working with a robot as a digital manipulative in mathematics teaching allows students to develop a deeper mathematical understanding by contributing to their reasoning, problem-solving, generalization, and estimation skills [46]. On the other hand, according to Lee et al. (2011), teachers do not have the opportunity to access the necessary materials and technology to develop their CT skills as a part of their professional development (as cited in [47]). According to Finsterbach Kaup (2020), professional development courses given to teachers from time to time are not sufficient. In addition, there is a need to establish professional development practice communities with continuous support and resource sharing.

D. *Digital Competence and its Relationship with CT*

Digital competence is defined as "the ability to understand and use the information presented in various formats from different sources through a computer" (Knobel & Lankshear, 2008, cited in [48]). The definition of digital competence, which is shown by the Council of the European Parliament as one of the eight basic competencies that citizens must have to adapt to the changing world conditions, has been made as follows. "Digital competence involves the confident and critical use of information society technologies for work, leisure, and communication. Digital competence is grounded by basic skills in information and communication, i.e. the use of computers to retrieve, evaluate, store, produce, present and exchange information, and to communicate and participate in collaborative networks via the Internet." [49]. According to this definition, digital competence, as a combination of knowledge, skills, and attitudes, requires deep knowledge and understanding of the nature, role, and possibilities of information technologies, not only in business but also in interpersonal and social life [49].

Digital competencies require extensive skills to master and apply technologies in the context of problem-solving, which is closely related to CT, beyond being familiar with specific technologies [50]. CT aims to develop students' potential in solving poorly structured problems by enabling them to perform a set of computational applications based on computational logic [50]. CT is associated with a range of thinking skills such as computer problem solving, algorithmic design development, and mathematical thinking [19]. England has included CT and coding in their curriculum in primary and secondary schools [40]. England claims that "A high-quality computing education equips students to use computational thinking and creativity to understand and change the world". England's pioneering is not just directed towards computation, but also in the conceptualization of CT, which is strongly promoted in its agenda for compulsory education [40].

Many countries in Europe have planned to include CT in compulsory education [40]. For example, in the Czech Republic, "Developing computational thinking among students" has been identified as one of the priority objectives in the Digital Education Strategy document. Therefore, CT is seen as a key to the acquisition of digital competencies that all students need to understand their future lives, professional careers, and the world [40]. In Scandinavian European countries, the concept of "digital bildung" is used to express what it means to be literate in contemporary culture. In the Norwegian national curriculum, digital competence is defined as a complex competence that reveals simple ICT skills (using software to search, transform and control information) and more complex skills such as the use of digital tools and media [51].

Information communication technology applications such as Web 2.0 tools, multi-user virtual environments and augmented reality contribute to the development of 21st-century competencies (Dede, 2010a, cited in [51]). For example, Web 2.0 technology enables users to create and share content in new ways. The digital camera and different software tools make it easier to show and mirror students' work. Immersive interfaces such as multi-user virtual environments and augmented realities offer two powerful opportunities to learn 21st-century competencies (Dede, 2010a, cited in [51]). First, it enhances digital competence and the ability to access distance and time-dispersed sources of information and the psychosocial community. Second, it develops the ability to create activities that are not possible in the real world.

According to Dede (2010b), digital competence should not be regarded as a separate skill set but should be placed within and among other 21st century competencies and basic subjects (as cited in [51]). Bennet and Maton (2010) state that due to the widespread use of technology in society, schools and universities generally assume that students have digital competence, but there are great differences in students' use of technology and thus technology skills, and this situation becomes clearer day by day (as cited in [51]).

Most frameworks describing twenty-first-century competencies state that new evaluation frameworks are needed to evaluate 21st-century competencies. It is stated that performance evaluation strategies are necessary to understand how students progress in mastering 21st-century competencies [51]. Voogt et al. (2013) emphasize that the standards to be established for digital competence should not lead to fixed tests and evaluations and should be flexible depending on the speed of technological and cultural change. It is also stated that it is important to see the use of technology in real life (Lankshear & Knobel, 2006, cited in [51]), and to understand such processes, it is necessary to look at the different contexts in which technology literacy is applied and made sense, and how new technologies change the nature and processes of meaning-making. This is especially important in relating how children and young adults use digital technologies [51]. It is necessary to follow the studies of countries dealing with the development of new evaluation frameworks and the development of evaluation tools for the evaluation of 21st-century competencies. Because these studies can be a guide for countries that want to strengthen their evaluation studies on a national basis (Dede, 2010b, cited in [51]).

E. *The Twenty-First-Century Learning Environment and its Importance in the Development of CT*

The environment formed by the interaction of space, time, infrastructure, equipment, and psycho-social factors in the learning process and affecting this process is called the learning environment (Acat, 2005, cited in [52]). Learning environments are environments where individuals use existing resources for a purpose to explain the events in the environment and develop solutions to the problems encountered (Wilson, 1996, as cited in [52]). The term “learning environment” refers to the place and space - it could be a school, a classroom, a library. Much of 21st-century learning takes place in physical settings like this. But in today’s interconnected and technology-driven world, a learning environment can be virtual, online, and distant; It doesn’t have to be any space.

CT has become a trendy word that promises to educate the new generation of children with a much deeper understanding of our digital world, and the educational environment is changing fast and we are now at a tipping point [40]. Moreover, the increasing interest in 21st-century skills has focused attention on the importance of smart learning environments in developing students’ digital competencies. The State Educational Technology Directors Association [SETDA] (n.d.) defines 21st-century learning environments as classrooms with computer hardware, software, electronic whiteboards, and rich digital and online curricular resources. When entering such an environment, it is noticed that interactive learning, higher-level thinking skills, and student engagement are pervasive. Through the programs offered in such environments, students are provided with the opportunity to access and collaborate with the relevant content they need most and often cannot access. To create 21st-century learning environments, it is necessary to make collaborative planning, investment in core components for technology, and give importance to intensive professional development and training of teachers. In this context, elements such as not only how to use technology in the training of teachers, but also how to integrate technology into the curriculum, determining the content of the curriculum, providing rich digital content, and CT support are the key elements that need to be considered in the process of transforming schools [53]. CT needs to be included in formal education to give all children equal opportunities and to provide them with the CT skills they need. This will only deliver results if policymakers set out their vision and carefully define, plan and monitor their concrete implementation steps [40]. While experts emphasize that it is important to introduce children to CT concepts in the early stages of school, they state that more studies should be done on what is the most appropriate age [40]. However, the current general assumption is that it is appropriate to develop CT-related skills from an early age. Therefore, new comprehensive approaches are needed to cope with the complexity of cognitive processes related to CT [40]. On the other hand, the organizers of learning environments are teachers. Ensuring the continuous professional development of teachers will enable the development of learning environments and the analysis of teaching and learning processes in the best way [46]. In this process, it is necessary to determine new tools and criteria for how teachers can evaluate their CT skills. The inclusion of CT in their curriculum creates a strong demand for continuing professional development (CPD) for teachers who do not have sufficient knowledge of this concept. It is of paramount importance that teachers and school staff should be provided with training opportunities that strongly focus on CT pedagogy and hands-on learning which can be easily transferred to the classroom [40].

F. *Opinions on Evaluation of CT*

Papert (1980) emphasizes in his studies that computer culture can shape children's thinking and that they can learn by programming the computer and coming into contact with strong ideas. This idea encourages studies on the transfer of cognitive skills from programming activities to other fields, so the idea that cognitive skills such as problem-solving skills can be transferred is seen as a necessity to include CT in the compulsory curriculum [40]. Evaluation of CT concepts and applications is important to see the development of CT. Multiple evaluation mechanisms, including qualitative data, are needed for the evaluation of CT skills. Evidence of CT-related behaviors was observed by researchers and CT was evaluated using various methods [50]. Moon et al. (2020) proposed a model to facilitate and develop the acquisition of CT skills using the Open Educational Resource (OER) as one of the ways to support personalized learning. However, the researchers stated that current OER designs and applications are limited in improving students' CT skills and that design-based research is needed to adapt OERs to improve CT. It has been emphasized that it is important to know how to reduce the increased cognitive load of students and how to coordinate practices in instructional design [50]. Hosseini (2017) evaluated students' programming behaviors using the evidence-centered design (ECD) framework in the development of CT, and classified students' main observable behaviors regarding programming skills as previous programming experience, intrinsic motivation levels, and math and science skills (as cited by [50].

Another example of using multiple evaluation mechanisms is The Computational Thinking Pattern Analysis (CTPA) framework. This framework has produced early indicators of transfer from game design to computational science [40]. The basic assumption of the CTPA framework is that students can develop games using CT patterns and apply the same patterns in other areas. This framework breaks down complex programs, showing ways to measure these patterns in student-created products, and provides evidence of skill transfer across fields.

Brennan and Resnick (2012) offer three main approaches for assessing the development of CT. These approaches can be listed as follows:

- Analyzing students' project work and generating visual representations of the programming used or not used in each project,
- Discussion of generated products,
- Design scenarios - given a set of three projects with low-medium-high complexity levels; explaining the pu-

-rpose of the selected project, describing how it can be extended, fixing a bug, and remixing the project by adding a feature.

Many strategies suggest assessing CT by analyzing the artifacts (e.g. games or models) that students develop. It is stated that troubleshooting scenarios can be considered as an effective way to assess students' fluency in computer programming and computer-based problem solving [40]. Various tools are being developed to support educators' evaluation of students in programming and evaluate the development of CT [40]. One of them is Dr. Scratch, a tool that performs automatic analysis of Scratch programs and detects the presence or absence of certain key statements in students' work, such as conditional statements. Wilson, Hainey, and Connolly (2012) developed a coding scheme consisting of design categories for programming concepts, code organization, and usability to evaluate students' CT skills while using the Scratch program. Wiggins, Grafsgaard, Boyer, Wiebe, and Lester (2017) mostly used JavaTutor, a special computing platform that takes into account the cognitive and affective states of students. On the other hand, while it is stated by Gal-Ezer, Peyton-Jones, Lepeltak, Urschitz, and Voogt that current evaluation methods and tools cover some aspects of CT, it is emphasized that the progress in the evaluation of CT is not at the desired level, therefore more research is needed [40].

III. Discussion

In today's world where scientific and technological developments are developing rapidly, it becomes important to raise individuals with the skills required by the 21st century. Because the security, health, and economic policies of nations are significantly affected by the developments in digital technologies. Especially developed and developing countries regulate their current and future employment policies according to technological developments [1].

With the development of Computer Science, the interest in computational thinking, which was previously thought to be completely related to ICT, but which was redefined by each discipline in line with its own needs, and therefore accepted as a much more comprehensive and problem-solving method than computer science, has increased. With the understanding of CT, it becomes easier to solve problems, design systems, and understand human behavior [3]. From this point of view, CT is a skill that must be acquired by everyone, being one of the 21st-century competencies ([1], [3], [4], [54]).

Increasing interest in CT has led to various studies. This study focused on the core components of CT and its relationship with other competencies and digital competence in the 21st century, the importance of integrating it into compulsory education and the challenges faced in this process, the proposed pedagogical approaches, and related issues based on teacher training, computation-computational theory, mathematical sciences, the expansion of mathematical sciences over time, the relationship of computer science with CT, various reports ([11], [13], [14], [15], [55], [56]), and studies (ex. [1], [3], [4], [16], [57], [58], [59]) are included, and original results are obtained, and conceptual frameworks are determined. In this context, the results obtained are given as items, and evaluations are made.

A. *Evaluation of Results from Computation, Computational Theory, CSE, and Mathematical Sciences' Relationship to CT*

While CT uses the basic concepts of computation and computer science [3], it is also at the core of modeling, reasoning, and problem-solving in many mathematical disciplines ([5], [6]). The steps of the computation, which is a process defined by the computational model, are used in the generation of the solution algorithms of the problems in CT. Mathematical abstractions called computational models are at the core of computation and CT ([7], [16]). Considering that the theory of computation allows research based on determining whether a problem in computer science can be solved with certain algorithms and computational models, how long it can be solved if it is solved, and how many resources are needed ([60], [7]), it turns out that computation and computational thinking cannot be considered separately from each other. Because CT has been defined based on abstraction, decomposition, and error analysis necessary for solving complex problems, designing systems, and understanding human behavior by using the basic concepts of computer science ([1], [3], [4]). In addition, the fact that the theory of computation is one of the main areas of applied mathematics [7] clearly shows the relationship between computation and computational thinking. The application of mathematical knowledge to other fields to solve real and complex world problems is the field of applied mathematics. Computational thinking also uses computational and computer science concepts to solve complex problems and incorporates a range of skills such as mathematical modeling, reasoning, and problem-solving. It is understood that computational theory and computational thinking are related to each other in the context of finding solutions to real-world problems, common concepts used and the path followed. Computing informs societies and policymakers in a wide variety of fields such as climate change, public health, and the environment [10]. Computational thinking is also defined as a comprehensive problem-solving skill. For this reason, solution algorithms and computational models required for solving real-life problems faced by societies can be created using CT dimensions. The existence of computational tools and computational models (ex. Turing machine, Pascal's and Leibnitz's mechanical calculating machines, Napier's logarithms) that enable digital technologies to reach today's developments support the statements given above.

The importance of CSE has increased significantly in the new century, as CSE is a discipline in which research is conducted on the development and use of computational methods to enable scientific discovery to support innovation in engineering and technology and decision-making in socially important applications areas. While the developments in this field direct the theoretical and empirical scientific and technological developments, they have an impact on the planning of government policies in every field that concerns the society ([11], [12].). The relationship of CSE with the fields of mathematics and statistics, computer science and science, and engineering has led to the emergence of CSE as a unique new field. The effort of the CSE community to produce effective solutions to complex real-world problems, either theoretically or empirically, supports the definition of computational thinking in this context. The fact that CSE is a unique new field, combining mathematical sciences, computer sciences, science, and engineering sciences, is related to the use of concepts in these fields. Therefore, the underlying relationship of the CSE community with computational thinking is visible.

Studies in the field of mathematical sciences, which use the interactive integration of mathematics, statistics, and computation, show themselves in many fields such as biology, medicine, social sciences, business, climate, and finance. The reason for this is that the world of science encounters new problems every day in many fields and the solutions to these problems cannot be done without associating them with mathematics [14]. Many areas of CSE, where mathematical sciences are natural languages, are based on computation and simulation. The science and engineering process based on simulation is mathematical and is based on mathematical modeling. Mathematical sciences transition between fields called basic and applied mathematics, statistics, operations research, and theoretical computer science, and therefore the boundaries between the fields of mathematical sciences are not drawn [13]. The fact that an idea in any field affects another and helps to produce different ideas has caused the mathematical sciences to expand over time to reveal different fields [14]. On the other hand, the spread of computation in recent years has transformed mathematical sciences and application areas. Accordingly, computational science provides more and more services to science by enabling a virtual laboratory where science is applied [15]. Computational science allows calculations that require high precision, and the results can be predicted by simulation. To do this, a computational model takes programming structures from computer science and methods and theories from mathematical sciences. Thus, computational science synthesizes its unique knowledge from these two fields and uses it in generating and testing engineering designs [15].

In summary, the following evaluation can be made. Computational sciences use computational and computational theories to synthesize and test information from mathematical and computer sciences in generating and testing science and engineering (CSE) designs. The designs are generated by taking programming structures (algorithms) from computer sciences and using mathematical models developed with mathematical structures from mathematical sciences. Meanwhile, the designs are tested with mathematical simulations and the results are predicted. Mathematical sciences, computer sciences, and computational sciences make various contributions to each other in the generation process from the formation of a design idea to the design and testing the results and determining the prediction. Therefore, it is understood that an idea in one field helps to develop and put into practice another idea in another field ([11], [15]) and this reveals the relationship between these fields.

B. *Evaluation of Results Arising from the Relationship between Computing Principles and CT Principles*

CT involves defining, understanding, and solving problems, reasoning at multiple levels of abstraction, understanding and applying automation, and analyzing the appropriateness of abstractions [18]. CT is thus a comprehensive set of approaches and skills with applications in various disciplines. CT expands skills related to different thinking areas such as algorithmic thinking, engineering thinking, design thinking, and mathematical thinking, so it shares common skills with these thinking areas. These skills are defined as abstraction, problem-solving, analysis, innovation, and automation [18]. The computing principles of computation, communication, coordination, recollection, automation, evaluation, and design [20] make important contributions to the generation, understanding, organization, and categorization of CT in science fields other than computer science [61]. On the other hand, Brennan and Resnick (2012) state that computational concepts are not sufficient to explain CT, so computational applications should be conducted. Because computational applications focus on how students learn rather than what they learn. Students use skills such as testing, debugging, abstracting, and modularizing their designs while practicing. Moreover, students have the opportunity to see the real-life reflections of their products, to establish connections, to question, to introduce their products, and to express themselves. The model proposed by Brennan and Resnick (2012) is important in terms of showing how computing and CT principles work together and contribute. Similarly, Chiasson (2019) proposed a model that shows how students use theoretical concepts in the CT process and how this process contributes to students. It is thought that the models proposed by Brennan and Resnick (2012) and Chiasson (2019) will guide experimental and theoretical research for the development of CT in education.

C. *Evaluation of CT as a Pedagogical Device*

According to the results from studies and reports (ex. [18], [24], [40], [41], [42]) conducted in recent years, it is understood that the development of CT in compulsory education depends on the provision of teacher competence, the revision of curricula, the creation of effective and adequate pedagogical frameworks, the identification and resolution of the difficulties encountered in practice. CT skill is becoming the new literacy of the 21st century. Therefore, structured pedagogical frameworks are needed to acquire this skill [18]. To facilitate the development of computational thinking, CT was defined as a pedagogical device and formed as the fields of "production", "reconceptualization" and "reproduction" ([9], [22], [62]). It is important to focus on the "reconceptualization" dimension of CT as a pedagogical device because it is in this dimension that knowledge creation takes place [18]. To assist in the development of CT, many researchers ([9], [18], [26], [25], [64], [65], [63]) have developed various pedagogical models. The common feature of the developed models is that they focus on finding methods that will facilitate the learning of CT in all disciplines. These models predict that students will acquire skills to solve real-life problems. Researchers suggest that a computing-based model that promotes STEM learning can be developed in innovative and collaborative teaching environments [65] and that CT skills can be assessed by establishing an appropriate program design and analysis framework [64], however, they state that the realization of all this depends on the introduction of CT.

According to Philips (2009), CT enables the creation of knowledge rather than using it, thus creating opportunities for creative solving of problems and facilitating the discovery of new ways. The model suggested by Chande (2015) and the positive results with the application of the model support Philips (2009) definition of CT. In a model that includes four pedagogical experiences based on constructivism [25], it is stated that students' learning potentials are revealed while working and sharing what they do, and this situation will contribute to the creation of digital citizenship.

The difference of the model [26], in which creative teaching methods, unplugged approaches, and low-tech prototyping methods are used, from other models, is that it emphasizes the importance of imagination in design. In the model, it is stated that creativity can be developed with low-tech and information-rich learning environments based on CT. In this context, the importance of developing students' problem-solving skills and approaching the solutions of problems like designers is emphasized.

One of the applications that are thought to be beneficial in learning the basic concepts of CT is "unplugged activities" without using a computer. In cases where computers can be distracting from time to time or are not motivating enough, it is recommended to perform various activities without using computers to learn the basic concepts of CT ([29], [35], Campos et al, 2014, cited in [29]). However, using "unplugged activities" alone in the development of CT is not enough. It is recommended that such activities be used mostly in the initial stage and in combination with different pedagogical devices in learning CT concepts [35] and that more research should be conducted on younger students who will encounter CT for the first time [39].

As a result, based on the idea that "computers are not necessary" in learning CT, UP activities, which have been revealed by studies that contribute to the teaching of computer science concepts by using devices other than computers, should not be considered as the only way to learn CT. It should be used in conjunction with different machines, devices, and pedagogical devices, computational or not. It is thought that this association and the alternate structuring of studies in this context will contribute to the development of CT.

In this study, pedagogical models developed for the development of CT as a pedagogical device are introduced by emphasizing their remarkable features. However, for the solution of basic problems such as how to integrate CT into the curriculum in compulsory education, from which age group it is appropriate to start CT education, how to evaluate skills, how to do teacher training, and what difficulties may be encountered in practice, these problems should be discussed and further research should be conducted ([24], [40], [65]). Since the importance of integrating CT into education and the role of CT in education have been recognized by the world's societies in just recent years, it is seen that research on this subject is limited. Therefore, it is clear that more research is needed to see the effects of CT in practice and to identify the deficiencies. It can be suggested to use these models in experimental studies to be suitable for the study. However, it is considered that knowing its dimensions, proposed models, and frameworks as a pedagogical device for the development of CT alone is not enough. As it is known, the quality of teaching is determined by the teacher’s knowledge of knowing the student, knowledge of the curriculum and teaching methods, technology knowledge, beliefs, attitudes, perspective on teaching, in short, pedagogical and technological pedagogical content knowledge ([44], [66]). For this reason, it can be said that no matter how good a model is, if the teacher’s perspective and knowledge are not sufficient, the desired efficiency cannot be obtained from that model. For this reason, teachers' perspective on CT, which is a comprehensive problem-solving method accepted as one of the basic skills of the 21st century, uses computer science concepts and techniques and requires basic skills such as problem-solving, creativity, and critical thinking, gains importance in this context. Philips (2009) made various suggestions to teachers regarding the development of CT and in-class practices. According to Philips (2009), the essence of CT is to think about data and ideas and to combine these resources to solve problems. In the process of creating knowledge for the development of CT, teachers should encourage their students to use the dimensions of CT such as abstraction, problem-solving, computer concepts, and new technologies, and to know how to apply technology to reach and use information [30].

In the JRC (2016) report, it is stated that despite the widespread interest in CT in academic circles, societies, and government policies, there are various problems in integrating CT into compulsory education [40]. There are unsolved issues and challenges, especially on how to integrate CT into curricula, how to create learning environments equipped with digital technologies in the new century [21], and how to do CT training of teachers. In high-tech information societies, students need to develop their skills in science, technology, engineering, and computational mathematics. For this reason, countries that have included CT in their curriculum give importance to the teaching of CT and related concepts, regardless of whether CT teaching poses a social challenge or the employment conditions are appropriate. In addition, teacher training programs are also emphasized to support the integration of CT and CS into the curriculum in these countries. Countries focused on the development of students’ CT skills (Asia Pacific Region and European Union countries) primarily plan to use innovative pedagogical approaches specific to CT in their curriculum, to develop assessment tools to measure students’ CT knowledge, skills, attitudes and perceptions, and especially to train their teachers on this subject. It is seen that they continue their progress in these subjects by using innovative approaches. In addition, these countries use communication and cooperation approaches in their studies. It is stated that studies conducted in these countries in this context [1] contribute to the development of CT. On the other hand, studies show that teachers face various difficulties in the development of CT ([2], [31], [41], [42], [43], [67]). Moreover, the results of international studies have revealed that teaching strategies for the development of 21st-century competencies are often not adequately implemented [51]. However, limited studies show that with innovative pedagogical approaches to be applied and continuous professional development, the difficulties faced by teachers will be eliminated and positive results can be obtained ([42], [45], [46], [68], [69], [70]). The issues encountered in the development of CT and the provision of teacher proficiency and solution suggestions are listed below according to the results obtained from the researches.

- The use of real-life problems in CT applications helps the development of CT due to its contribution to the problem-solving process. Therefore, a curriculum designed around real-life problems should be prepared [41].
- A holistic design approach should be taken as a basis in the curriculum (Van Merriënboer & Kirschner, 2007, cited in [41]).
- The lack of teachers to implement CT should be eliminated and practical difficulties encountered in curriculum implementation should be eliminated ([67], Wong et al, 2016, cited in [1]).
- Teachers should be informed about CT, ICT, and pedagogical approaches, and studies should be conducted to understand how these three are related [40]. A structure consisting of CT, programming and mathematics teaching should be in the curriculum [68].
- Wong et al. (2016) stated that due to the complex nature of CT, teachers have difficulties in understanding and explaining this concept, so it is important to eliminate this deficiency in teacher education (cited in [1]). CT should be integrated into curricula in a way that teachers can understand and apply ([31], [42]).
- Studies should be conducted on teachers' beliefs that they can understand and apply CT (Curzon et al, 2009, cited in [42]). Teachers listed the factors affecting their self-confidence and beliefs. (a) not having enough time to learn, (b) having difficulty in accessing relevant technological resources and needing more support in case of accessing them, (c) needing to meet and maintain communication with colleagues and field experts, (d) providing more examples, activities, and ideas for teaching CT [42].
- Teachers should be informed about the innovative processes and pedagogical approaches for the development of CT simultaneously with the developments, they should be made to feel included in the process, and their continuous professional development should be ensured [40].
- Unplugged activities should be used alternately and together with different pedagogical devices [35].

Although there are various difficulties in the development of CT, there are studies that offer various suggestions for the development of CT skills. The applications of information communication technologies such as Web 2.0 tools, multi-user virtual environments, and augmented reality, and the use of digital cameras and different software tools contribute to the development of CT skills (Dede, 2010a, 2010b, cited in [51]). To ensure teacher competence in the development of CT, studies ([2], [41], [42], [71], [72]) with positive results are conducted by the researchers, and models are suggested. It is thought that the issue of teacher competency in the development of CT can be solved by considering the results of these studies and applying the models.

D. *Evaluation of the Results from the Relationship of Digital Competence and New Century Learning*

Environments with CT

One of the five components of digital competence, defined as the ability to understand and use the information presented in various formats from different sources via computers (Knobel & Lankshear, 2008, cited in [48]), is problem-solving [73]. In this context, it can be said that the development of students' ability to use their digital competencies depends on the development of problem-solving skills, which are closely related to CT skills. In addition, CT is closely related to thinking skills such as computer problem solving, algorithmic design development, and mathematical thinking [19]. It can be concluded that the definitions of digital competence and CT support each other, and therefore an improvement in CT will lead to the development of digital competence, and an improvement in digital competence will lead to the development of CT. This result is supported by the fact that many European Union and Asia Pacific countries, which have set the development of CT as one of their primary objectives and have included CT in their curriculum, are united in the fact that CT is a key skill for students' digital competence acquisition [40]. Implementation of information communication technologies such as Web 2.0 tools, multi-user virtual environments and augmented reality offers various opportunities for the development of digital competence. Tools such as digital cameras and different software tools make it easier to show and reflect students' work (Dede, 2010a, cited in [51]).

Results from international studies show that teaching strategies for 21st-century competencies are often not well implemented in real educational practices and there are various difficulties. Problems such as the lack of integration of twenty-first-century competencies into curriculum and assessment, inadequate teacher preparation, and the lack of systematic consideration of the strategies necessary to adopt innovative teaching practices on a large scale can be counted among the reasons for this situation (ex. [51]). Some suggestions have been made by researchers to overcome these difficulties (Dede 2010b, cited in [51]). Presenting models and frameworks in which subject areas are associated with 21st-century skills, addressing 21st-century skills together, considering the potential of informal education environments in the development of these skills, creating new assessment frameworks, and recognizing the importance of countries and cultures in the development of these skills are among the main suggestions offered. In addition, standards and assessments for digital competence should be flexible depending on the speed of technological and cultural change, and the use of technology should be made in the context of real life. Because this is how children and young people use digital technologies to associate 21st-century skills (Lankshear & Knobel, 2006, cited in [51], [51]).

Today's technological developments provide the opportunity for learning environments to be virtual, online, and distant. It may be thought that there is no need for a physical space in learning environments, but 21st-century learning is generally done in physical environments. For this reason, the reorganization of teaching environments is of great importance in the development of CT. Because it does not seem possible by the definition of CT to carry out studies based on CT in traditional learning environments. CT will not accept traditional classroom settings and the passive learner. SETDA (n. d.) defined new century learning environments as classrooms where students' higher-order thinking skills are developed, student participation is ensured through student-centered approaches, and computers, electronic whiteboards, and rich digital and online resources are available. Chiasson (2019) conducted a small-scale study of the image of learning environments and school models that include new technologies. Chiasson's (2019) study provides important information about how new century learning environments should be. With this study, Chiasson (2019) showed that new learning and teaching approaches and learning environments equipped with new technologies in which students are active, create, verify and share their knowledge contribute to students' CT skills. It can be said in the context of studies that it is necessary to conduct similar studies and evaluate their results comparatively, to organize schools by the requirements of the age in this new century, and to put new student-centered pedagogical approaches into practice.

E. *Evaluation of the Conclusions Drawn from the Opinions Regarding the Evaluation of CT*

Evaluation of CT applications is important in terms of seeing the development of CT. Multiple assessment mechanisms are needed to see the extent to which CT gains are achieved. Researchers have suggested various assessment approaches in this regard ([40], [60], [74], [75]). Many assessment strategies suggest student-centered approaches and state that it would be appropriate to make an assessment based on games and models created by students. On the other hand, it is emphasized that the progress in the evaluation of CT is not at the desired level, so more research is needed [40].

As a conclusion of this study, which is based on the literature review, it is understood that the development of CT in compulsory education depends on the generation of effective and adequate pedagogical frameworks, the identification and elimination of difficulties in implementation, and its adoption as a government policy. To understand CT, it is necessary to focus primarily on computing concepts, computational theory, computer science concepts, and the mathematical sciences that form the basis of computer science by its definition. Because it can be said that the development of CT can be realized by knowing the dynamics that form the basis of CT and adopting a new understanding in an integrated way. With the understanding of CT, the second main point to focus on is how to integrate CT into compulsory education. In this context, it is important to answer basic questions. For example, while it is generally accepted that it would be appropriate to introduce students to CT from an early age, experts recommend that more research be conducted on this subject [40]. The questions, such as "What age group is CT suitable for students? How will teachers who do not know enough about this concept apply CT, which pedagogical approaches will be used, and how? How will the learning environments required by the twenty-first century be generated? How will technology teaching and digital competence be provided? How will politicians do the work necessary to develop 21st-century competencies and provide current and future employment?" need to be answered.

The main conclusions and recommendations to be drawn from all the explanations made so far can be expressed as follows:

- When the studies on CT, in general, are examined in the literature, it is seen that the common recommendations are generally to introduce CT, to gain CT skills, to include CT in teacher training programs and compulsory education, and to conduct more studies on its implementation.
- CT should be included in compulsory education to provide students with CT skills and to provide equal opportunities for all students. However, for this, politicians should consider this issue in their education policies, plan the process and monitor the process by taking into account all the parties and components of the issue [40]. In this process, the results of the studies and the opinions of the researchers on subjects such as the handling of CT skills together with other 21st century skills, the age group students will start with, the integration of CT into the curriculum, and the implementation of continuing professional development

(CPD) programs for teachers should be taken into account and should be applied mindfully.

- Different definitions of CT in different disciplines have led to different interpretations of this approach [76]. In this context, multiple CT models emphasize different aspects and dimensions of CT according to the disciplines it is defined in [76]. In recent years, with the demand to integrate CT into education, different experimental studies in different disciplines have started to be seen. Various models are proposed and studies are carried out to develop CT skills, especially in teacher training programs. However, it seems that these models, approaches, and practices are not explanatory enough to help students develop their CT skills [76]. It is thought that there are several reasons why applied studies in the fields of science, mathematics, engineering, and technology, which are thought to be closely related to CT, are not sufficient. Integration of CT into education and designing courses based on appropriate curricula, the inclusion of CT in teacher training programs, and the necessity of arranging the training of teachers on this subject through in-service training are among these reasons.
- Two of the important factors in the effectiveness of the teaching process are teachers' pedagogical content knowledge and their beliefs in learning-teaching. Clarke and Hollingsworth (2002) suggested that teachers' beliefs and knowledge are linked to their practices. In this context, new applications to be made with the inclusion of CT in teaching will increase the quality of teaching and contribute to the professional development of teachers by giving them opportunities to use new methods and strategies.

VI. Conclusion

The importance of CT has been increased in the 21st century with the development of digital technologies. Although there is no clear definition in the literature, it can be said that CT is an extended problem-solving method that combines human thought and technology. On the other hand, CT is more than a problem-solving method; and it is one of the 21st-century essential skills that every individual should acquire. Various reports written in many countries, especially in Europe and Asia, recommend developing problem-solving skills and 21st-century skills. Recognizing the importance of CT has led to the development and change of education policies. Many countries have changed and are changing their curricula to teach CT concepts and develop CT skills. At the same time, this situation has attracted more researchers' attention, and in this context, researchers have proposed various models. However, there are not enough studies in the literature on the outcomes of these models in practice. Various difficulties have been identified, especially in the practical applications of CT, and no definite way, method, and understanding could have been determined on how to solve these challenges. The issue of integrating CT into teacher education and curricula and measuring and evaluating it is still under discussion. In addition, the importance of 21st-century learning environments where digital competence in teaching CT concepts will be given was also emphasized. In some studies, the success and motivation of students in learning CT concepts have been observed to increase in learning environments organized as required by the 21st-century.

Even though there are some problems, including teacher training, development of CT, the regulation of curricula, and the determination of measurement and evaluation methods, it is clear that CT is a skill that needs to be gained in the 21st-century. There is a need for further studies on the acquisition of CT concepts and the development of CT skills regarding models and theoretical infrastructure in this context.

REFERENCES

[1] So, H.J., Jong, M. S.Y., & Liu, C.C. (2020). Computational thinking education in the asian pacific region. *Asia-Pacific Education Researcher, 29*(1), 1–8, https://doi.org/10.1007/s40299-019-00494-w.

[2] Grover, S. & Pea, R. (2013). Computational Thinking in K-12: A Review of the State of the Field. *Educational Researcher, 42*(1), 38-43. doi: 10.3102/0013189X12463051

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

[4] 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.

[5] Sengupta, P., Kinnebrew, J. S., Basu, S., Biswas, G., & Clark, D. (2013). Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework. *Education and Information Technologies, 18*(2), 351-380. https://doi.org/10.1007/s10639-012-9240-x

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

[7] Maheshwari, A. & Smid, M. (2019). Introduction of theory of computation. School of Computer Science, Carleton University, Ottowa, Canada. Retrieved July 1, 2020 from https://cglab.ca/~michiel/TheoryOfComputation/TheoryOfComputation.pdf

[8] Turing, A. (1937) On computable numbers, with an application to the entscheidung sproblem. *Proceedings of the London Mathematical Society, 2*(42), 230–265.

[9] Bernstein, B.B. (1990). Class, Codes and Control: Vol.4: *The Structuring of Pedagogic Discourse,* London: Routledge.

[10] Agreste, S. & Puccio, L. (2011). Wavelet-based watermarking algorithms: theory, applications and critical aspects. *International Journal of Computer Mathematics, 88*(9), 1885–1895.https://doi.org/10.1080/00207160.2010.502225

[11] Computational Science and Engineering [CSE] Report, (2017). Research and Education in Computational Science and Engineering, *Report from a work shop sponsored by the Society for Industrial and Applied Mathematics (SIAM) and the European Exascale Software Initiative (EESI-2),* August 4-6, 2014, Breckenridge, Colorado.

[12] Rüde, U., Willcox, K., Mclnnes, L. C., & Sterck, D. H. (2018). Research and education in computational science and engineering. *[SIAM] Society for Industrial and Applied Mathematics Review, 60*(3), pp. 707-754.

[13] National Research Council [NRC], (2013). *The Mathematical Sciences in 2025.* National Research Council, Washington, D.C.: The National Academies Press. www.nap.edu

[14] National Science Foundation [NSF], (1998). *Report of the senior assessment panel fort he international assessment of the U. S. mathematical sciences.* Arlington, Va.

[15] PWN [Platform Wiskunde Nederland], (n. d.).Formulas for Insight and Innovation, Mathematical Sciences in the Netherlands, Vision document 2025. https://platformwiskunde.nl/

[16] Aho, A.V. (2012). Computation and computational thinking. *The Computer Journal, 55*(7), 832-835.

[17] Selby, C., & Woollard, J. (2014). Refining an understanding of computational thinking. http://curriculumvandetoekomst.slo.nl/21e-eeuwse-vaardigheden

[18] Chande, V.S. (2015). A Conceptual Framework for Computational thinking as a Pedagogical Device. *International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: Certified Organization). 3,*(11), 11682-11688. doi: 10.15680/IJIRCCE.2015. 0311104

[19] Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads, 2*(1), 32-37. doi: 10.1145/1929887.1929902.

[20] Denning, P.J. (2003). Great principles of computing. *Communications of the ACM, 46*(11), 15-20. https://doi.org/10.1145/948383.948400

[21] Chiasson, M. (2019). The relationship between learning spaces and the process of computational thinking skill. AERA Online Paper Repository. doi: 10.302/1440741 http://www.aera.net/repository

[22] Singh, P. (2002) Pedagogising knowledge: Bernstein's theory of the pedagogic device. *British Journal of Sociology of Education, 23* (4), 571-582. doi:10.1080/0142569022000038422

[23] 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.

[24] Lu, J.J. & Fletcher, G.H.L. (2009). Thinking about computational thinking. ACM SIGCSE Bulletin, 41(1), 260-264. doi: 10.1145/1508865.1508959.

[25] Kotsopoulos, D., Floyd, L., Khan, S., Namukasa, I. K., Somanath, S., Weber, J., & Chris, Y. (2017). A pedagogical framework for computational thinking. *Digital Experiences in Mathematics Education,* doi: 10.1007/s40751-017-0031-2

[26] Tsortanidou, X., Daradoumis, T., & Barberá, E. (2019). Connecting moments of creativity, computational thinkin, collaboration and new media literacy skills. *Information and Learning Sciences, 120*(11/12), 704-722. doi: 10.1108/ILS-05-2019-0042

[27] Razzouk, R. & Shute, V. (2012). What is design thinking and why is it important? *Review of Educational Research, 82*(3), 330-348. doi: 10.3102/0034654312457429

[28] Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22*(1), 142-158. Retrieved from https://www.arntechib.org/p/204418/.

[29] Brackmann, C.P., Roman-Gonzalez, M., Robles, G., Moreno-Leon, J., Casali, A., & Barone, D. (2017). Development of computational thinking skills through unplugged activities in primary school. *In Proceedings of the 12th Workshop on Primary and Secondary Computer Education* (pp. 65-72). Nijmegen, Netherlands, November 8-10. New York, USA: ACM Press. https://doi.org/10.1145/3137065.3137069 .

[30] Phillips, P. (2009). Computational thinking a problem solving tool for every classroom. *Computer Science Teacher Association.* http://csta.acm.org/Resources/sub/ResourceFiles/CompThinking.pdf.

[31] Tikva, C. & Tambouris, E. (2021). Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature review. *Computers & Education 162,* 1-23. https://doi.org/10.1016/j.compedu.2020.104083

[32] Yen, J. C. & Liao, W. C. (2019). Effects of cognitive styles on computational thinking and gaming behavior in an educational board game. *International Journal of Learning Technologies and Learning Environments, International Institute of Applied Informatics, 2*(2), 1-10.

[33] Corradini, I., Lodi, M., & Nardelli, E. (2017, August). Conceptions and misconceptions about computational thinking among ıtalian primary school teachers. ICER '17: Proceedings of the 2017 ACM Conference on International Computing Education Research. doi: 10.1145/3105726.3106194

[34] Hunsaker, E. (2018). Teaching Computational thinking and Coding Education basics to Early Childhood Educators. Unpublished masters project manuscript, Department of Instructional Psychology and Technology, Brigham Young University, Provo, Utah. Retrieved from https://scholarsarchive.byu.edu/ipt_projects/8

[35] Aranda, G. & Ferguson, J., P. (2018). Unplugged programming: The future of teaching computational thinking? *Pedagogika, 68*(3), 279-292. http://pages.pedf.cuni.cz/pedagogika/
[36] Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *New Zealand Journal of Applied Computing and Information Technology, 13*(1), 20-29.
[37] CS Unplugged, (n.d). https://csunplugged.org/en/topics/searching-algorithms/sequential-and-binary-search-unit-plan/how-many-guesses-junior/
[38] AlAmer, R.A., Al-Doweesh, W.A., Al-Khalifa, H.S., & Al-Razgan, M.S. (October, 2015). Programming unplugged: Bridging CS Unplugged activities gap for learning key programming concepts. *Proceedings of the Fifth International Conference on e-Learning* (pp. 97-103). Manama, Bahrain.
[39] Tsarava, K., Moeller, K., & Ninaus, M. (2018). Training computational thinking through board games: The case of crabs. *International Journal of Serious Games, 5*(2). https://doi.org/10.17083/ijsg.v5i2.248
[40] Bocconi, S., Chioccariell, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing Computational Thinking in Compulsory Education: Implications for policy and practice*. *JRC Science for policy report* Eds. (P. Kampylis, Y. Punie). Luxembourg: Publications Office of the European Union. doi:10.2791/792158
[41] Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 computational thinking curriculum framework: ımplications for teacher knowledge. Educational Technology & Society, 19 (3), 47–57.
[42] Bower, M., Wood, L.N., Lai, J.W., Howe, C., Lister, R., Mason, R., Highfield, K., & Veal, J. (2017). Improving the computational thinking pedagogical capabilities of school teachers. *Australian Journal of Teacher Education, 42* (3), 53-72. http://dx.doi.org/10.14 221/ajte.2017v42n3.4
[43] Sentance, S., & Csizmadia, A. (2015). Teachers' perspectives on successful strategies for teaching. *Computing in school. Paper presented at IFIP TCS 2015*. http://community.computingatschool.org.uk/files/6769/original.pdf
[44] Shulman, L. S. (1987). Knowledge and teaching: Foundations of the new reform. *Harvard Educational Review, 57*(1), 1-22. https://doi.org/10.17763/haer.57.1.j463w79r56455411
[45] Reichert, J.T., Couto Barone, D.A., & Kist, M. (2020). Computational thinking in K-12: an analysis with mathematics teachers. *EURASIA Journal of Mathematics, Science and Technology Education, 16*(6), 1-14. https://doi.org/10.29333/ejmste/7832
[46] Finsterbach Kaup, C. (September, 2020). Examining educational staff's expansive learning process, to understand the use of digital manipulative artefacts to support the students' computational thinking and mathematical understanding. pp. 53-60. *Proceedings of the Tenth ERME Topic Conference (ETC 10) on Mathematics Education in the Digital Age (MEDA), Austria.* https://hal.archives-ouvertes.fr/hal-02932218
[47] Back, J., Brodie, J., Curzon, P., Myketiak, C., McOwan, P.W., & Meagher, L.R. (2013). Making computing interesting to school students: teachers' perspectives. *ITiCSE '13.*
[48] Larraz, V., & Esteve, F. (2015). Evaluating Digital Competence in Simulation Environments. E. G. Bullen içinde, *Teaching and Learning in Digital Worlds, Strategies and Issues in Higher Education (*s. 99-105). Tarragone: URV.
[49] European Council (2019). Recommendation on key competences for lifelong learning. European Commission, Education and Training.. doi: 10.2766/291008 https://ec.europa.eu/education/education-in-the-eu/council-recommendation-on-key-competences-for-lifelong-learning_en
[50] Moon, J., Do, J., Lee, D., & Woong Choi, G. (2020). A conceptual framework for teaching computational thinking in personalized *OERs. Smart Learning Environments, 7*(6), 2-19. https://doi.org/10.1186/s40561-019-0108-z
[51] Voogt, J., Erstad, O., Dede, C., & Mishra, P. (2013). Challenges to learning and schooling in the digital networked world of the 21st century. Journal of Computer Assisted Learning, 29, 403–413. doi: 10.1111/jcal.12029
[52] Kaplan, M. (2011). Ilkogretim öğrencilerinin fen ve teknoloji dersi ogrenme ve öğretme ortamına yonelik dusünceleri (Primary school students' thoughts on the learning and teaching environment of science and technology lessons). *Journal of Education and Humanities: Theory and Practice, 2*(4), 77-92.
[53] State Eductional Technology Directors Association [SETDA]. (n.d.). 21st century learning environment models, 1-3. https://files.eric.ed.gov/fulltext/ED522778.pdf
[54] Hadad, S., Shamir-Inbal, T., Blau, I., & Leykin, E. (2020). Professional development of code and robotics teachers through small private online course (SPOC): Teacher centrality and pedagogical strategies for developing computational thinking of students. *Journal of Educational Computing Research, 0*(0), 1–29. doi: 10.1177/0735633120973432
[55] International Society for Techonology in Education [ISTE], (2015). Computational thinking competencies. Retrieved June 25, 2020 from https://www.iste.org/standards/computational-thinking
[56] EDUsummIT (2013). Summary Report. (1 to 2 October). Washington D.C., USA.
[57] Barcelos, T. S., Munoz, R., Villarroel, R., Merino, E., & Silveira, I. F. (2018). Mathematics learning through computational thinking activities: A systematic literature review. *Journal of Universal Computer Science, 24*(7), 815-845.
[58] Priami, C. (2007). Computational thinking in biology. *Technical Report CoSBi 10,* 1-5.
[59] Hsu, T.-C., Chang, S.-C., & Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Computers & Education, 126*, 296–310. https://doi.org/10.1016/j.compedu.2018.07.00 4.
[60] Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In A. F., Ball, & C. A., Tyson (Eds.), *In Proceedings of the 2012 Annual Meeting of the American Educational Research Association* (pp. 1-25). Vancouver, Canada: AERA. Retrieved from http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf
[61] Settle, A. & Perkovic, L. (2010). Computational thinking across the curriculum: a conceptual framework. *Technical Reports. Paper 13,* 1-49. http://via.library.depaul.edu/tr/13
[62] Haugen, C. R. (2009). *Contextualizations and recontextualizations of discourses on equity in education.* PhD Unpublished Thesis. Norwegian University of Science and Technology, Trondheim.
[63] Papert, S. (1980). *Mindstorms. Children, computers and powerful ideas.* New York: Basic Books.
[64] Malyn-Smith, J. & Lee, I. (2012). Application of the occupational analysis of computational thinking-enabled stem professionals as a program assessment tool. *Journal of Computational Science Education, 3*(1), 1-9. Retrieved September 6, 2020 from https://pdfs.sem anticscholar.org/12ca/bbdf849a848b53bde7f00a99a16452ebbbd2.pdf
[65] Pontelli, E., Brown, S., Cook, J., Sanchez, S., & Nava, C. (2016). New, GK-12: Computing in Context: Advancing Computational Thinking in the Classroom through Applied Computational Research. *Grant number: 0947465, Research organization: Laura Regassa, Division of Graduate Education [DGE],* New Mexico State University, United States.
[66] Shulman, L.S. (1986). Those who understand; Knowledge growth in teaching. *Educational Researcher, 15* (2), 4-14. https://doi.org/ 10.3102/0013189X015002004
[67] Peng, W. J., McNess, E., Thomas, S., Wu, X. R., Zhang, C., Li, J. Z., & Tian, H. S. (2014). Emerging perceptions of teacher quality and teacher development in China. *International Journal of Educational Development, 34,* 77-89. https://doi.org/10.1016/j.ijedu dev.2013.04.005

International Journal of Innovation and Research in Educational Sciences
Volume 8, Issue 5, ISSN (Online) : 2349–5219

[68] Gleasman, C. & Kim, C. M.(2020). Pre-Service teacher's use of block-based programming and computational thinking to teach. *Elementary Mathematics Digital Experiences in Mathematics Education, 6,* 52–90. https://doi.org/10.1007/s40751-019-00056-1

[69] Ketelhut, D.J., Mills, K., Hestness, E., Cabrera, L., Plane, J., & McGinnis, J.R. (2020). Teacher change following a professional development experience in integrating computational thinking into elementary science. *Journal of Science Education and Technology 29,* 173–187. https://doi.org/10.1007/s10956-019-09798-4

[70] Kong, S.-C., Lai, M., & Sun, D. (2020). Teacher development in computational thinking: design and learning outcomes of programming concepts, practices and pedagogy. *Computers & Education, 151*, 1-19. https://doi.org/10.1016/j.compedu.2020.103872

[71] Mishra, P. & Koehler, M. J. (2006). Technological pedagogical content knowledge: A framework for teacher knowlege. *Teachers College Record, 108*(6), 1017-1054.

[72] Mishra, P., Koehler, M.J., & Henriksen, D. (2011). The seven trans-disciplinary habits of mind: Extending the tpack framework towards 21st century learning. *Educational Technology, 11*(2), 22-28.

[73] Carretero, S., Vuorikari, R., & Punie, Y. (2017). DigComp 2.1 The digital competence framework for citizens. European Commission (2017).

[74] Wiggins, J. B., Grafsgaard, J. F., Boyer, K. E., Wiebe, E. N., & Lester, J. C. (2017). Do you think you can? The influence of student self-efficacy on the effectiveness of tutorial dialogue for computer science. *International Journal of Artificial Intelligence in Education, 27*(1), 130–153.

[75] Wilson, A., Hainey, T., & Connolly, T. (2012). Evaluation of Computer Games Developed by Primary School Children to Gauge Understanding of Programming Concepts. In P. Felicia (Ed.), *Proceedings of the 6th European Conference on Games Based Learning* (pp. 549-558). Academic Conferences and Publishing Limited (ACPIL).

[76] 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,* 147–166. https://doi.org/10.1007/s41979-020-00044-w

AUTHOR'S PROFILE

First Author
Neslihan Usta is currently Associate Professor in the Faculty of Education, Department of Mathematics and Science Science Education, Mathematics Education, Bartin University, Bartın, Turkey. Her research interests include mathematics education, teaching mathematics, teaching mathematics with games, problem solving, problem posing, problem-based learning, computational thinking, algebra. ORCID ID: 0000-0003-2662-1975. email id: neslihanusta74@gmail.com

Second Author
Seref Mirasyedioglu graduated from Ataturk University, Department of Mathematics and Physics in 1974. In 1976, he became an assistant at the same university. In 1979, he received a Doctorate in Computer Science. He conducted researches at Liege University between 1982-1983 with the scholarship of the Belgian Government and between 1985-1986 at TUBITAK Scholarship at Swansea University in England. In 1987, he received the title of Associate Professor in Applied Mathematics. Between 1987 and 1988, he conducted research with the British Royal Family Scholarship at Swansea University, England. In 1991, he was appointed to Hacettepe University Mathematics Department. Seref Miras--yedioğlu, who is the author of four books in the field of Applied Mathematics and Mathematics Education, is among the fields of study of mathematics education and computational thinking. He is still working as a Professor at Baskent University, Faculty of Education, Department of Mathematics Education and Director of Baskent University Distance Education Application and Research Center. ORCID ID:0000-0001-9492- 5992. email id: serefm@baskent.edu.tr