Id: 1201467

Participatory Educational Research (PER)
Vol.10 (2), pp. 1-25, March 2023
Available online at http://www.perjournal.com
ISSN: 2148-6123
http://dx.doi.org/10.17275/per.23.26.10.2

An Evaluation of the Effect of Activity-Based Computational Thinking Education on Teachers: A Case Study

Neslihan Durmuşoğlu SALTALI*
Necmettin Erbakan University, Faculty of Health Sciences, Child Development Department, Konya, Türkiye
ORCID ID: 0000-0002-6912-7080

Emel Bayrak ÖZMUTLU
Ordu University Faculty of Education, Primary Education Department, Ordu, Türkiye
ORCID ID: 0000-0002-1222-3557

Saniye Nur ERGAN
Ordu University Faculty of Education, Primary Education Department, Ordu, Türkiye
ORCID ID: 0000-0003-4782-7710

Gökhan ÖZSOY
Ordu University Faculty of Education, Primary Education Department, Ordu, Türkiye,
ORCID ID: 0000-0002-1250-624X

Özgen KORKMAZ
Amasya University Faculty of Engineering, Department of Computer Engineering, Amasya, Türkiye
ORCID ID: 0000-0003-4359-5692

Article history
Received:
08.09.2022

Received in revised form:
15.11.2022

Accepted:
15.12.2022

Key words:
Computational thinking, activity-based teaching, teacher education

This study aimed to conduct an in-depth evaluation of the activity-based computational thinking teaching practices performed to improve computational thinking and teaching skills of the basic education teachers. Based on the aim of the study, the case study design, one of the qualitative research methods, was selected. As a result of the collaborative work of five experts, a 20-hour education program built on two core competencies, four sub-competencies and eight thinking skills was implemented. The participants were 40 teachers, 20 of whom were classroom teachers and 20 of whom were pre-school teachers. Data were collected from three different sources using five data collection tools in order to conduct an in-depth analysis of the practices. Quantitative and qualitative data collection tools were used in a combined fashion in the research. The data were analyzed through content analysis and non-parametric analyses. Our findings revealed that thanks to the teaching practices performed, classroom teachers had significantly higher problem solving, diverse thinking, algorithmic thinking, and

* Correspondency: ndsaltali@gmail.com

computational thinking total scores, while preschool teachers achieved significantly higher total scores in algorithmic thinking skills and computational thinking. It was observed that the participants defined computational thinking on the basis of 18 different thinking skills. The explanations of the participants about the functions of computational thinking skills were grouped under seven categories. When the principles that should be considered in the teaching of computational thinking skills were examined, it was seen that the need for utilizing scaffolds was stated the most.

Introduction

Today, technology is considered as a new area of literacy rather than a set of tools that can be used to achieve existing learning goals. In this respect, digital literacy, which means using digital technologies correctly and effectively, is considered to be among the basic characteristics that individuals should have in this age (Angeli, 2021; Bocconi et al., 2016; Sari et al., 2022). Digital literacy can be thought of as a combination of many skills such as accessing information, analyzing information, producing, and sharing new information through technological tools and network devices (Park et al., 2020). The term computational thinking, coined by Papert (1980) within this broad skill network, is acknowledged to be one of the most important components of digital literacy.

It can be said that computational thinking is a problem-solving approach that uses technology and thinking together (Korkmaz et al., 2017). According to Wing (2008), computational thinking, which is a kind of analytical thinking, uses the same ways as mathematical thinking in problem solving, as engineering when designing and evaluating a complex system, and as scientific thinking in understanding concepts such as computability, intelligence, intellect, and human behavior. Similarly, Curzon (2015) defines computational thinking as a basic skill that targets problem solving and emphasizes the necessity of understanding what the problem is before thinking about solutions while solving a problem from a certain point of view. According to International Society for Technology in Education (ISTE, 2015), computational thinking includes creativity, algorithmic thinking, critical thinking, problem solving, communication and collaboration skills, and informatics people need to learn where, how and when to use digital tools to solve problems as a requirement of the information age. ISTE (2015) further states that the purpose of computational thinking in education is not to make students a leader in computer science, but to ensure that students are able to apply their computational thinking skills in other lessons as a habit. While the most important skill a person needs in problem solving is intelligence, developing the ability to solve problems with the help of computers and other digital tools has become one of the fundamental aspects of our daily life and work (Barr et al., 2011). In brief, according to ISTE (2015), computational thinking expands a person's problem-solving skills through computers and improves people's creativity and critical thinking abilities. Students use computational thinking when using algorithms to solve a problem and when solving problems with computation.

The process of computational thinking begins with data as input and with the search to derive meaning and answers from it. The output is not just a response, but the process of reaching it. It can be said that computational thinking draws the necessary route so that the process can be repeated, and others can learn from it and use it in order to draw a map for understanding. The drawing of this route includes four steps, which are **decomposition** that involves breaking the problem down into smaller, more manageable parts, **pattern recognition**, where the similarities and connections between different parts are identified by analyzing the data, **abstraction**, which involves identifying the information relevant to the problem and eliminating other unnecessary details, and algorithmic thinking, which is a process development phase that includes step by step solution to a problem so that the work can be repeated by humans or computers. The first three steps of computational thinking, namely decomposition, pattern recognition and abstraction, constantly feed the last step, algorithmic thinking (Hoyles & Noss, 2015; Rodríguez-Abitia et al., 2021; Shute et al., 2017). Understanding, testing, developing, or designing an algorithm refers to algorithmic thinking (Denning & Tedre, 2021). Wing (2011) defined the person with algorithmic thinking as an individual who is aware of different aspects of calculations in various problem situations and can solve these problems by applying the tools and techniques obtained from computer science.

In recent years, computational thinking, algorithm, and coding have attracted considerable attention around the world. Many countries have included activities that develop computational thinking skills in their education programs. These concepts have also attracted the attention of the scientific world. These developments have also had repercussions in the Turkish education system. In the vision document of Ministry of National Education-Turkey (MoNE) 2023 titled “Happy Children, Strong Turkey”, teaching of algorithmic thinking in classrooms in a computer-free environment has been given a special place since the basis of programming and coding is algorithmic thinking skills. Within the framework of the 2023 education vision, it is considered important that teachers are trained on game-based activities that support the algorithmic skills of children and they are introduced the educational materials that can be used in this regard. In this context, it is important that children acquire algorithmic thinking skills in a computer-free environment before they start using digital technologies. In the light of these developments all over the world, it has been emphasized that education on algorithmic thinking should be started from an early age, and thus, courses on programming and coding have been added to the curricula for different levels (Karabak & Güneş, 2013; Perry, 2009). However, it should be remembered that giving direct programming and coding education to children who cannot think abstractly at a young age can be challenging and boring for children (Lahtinen et al., 2005). Gomes and Mendes (2007), Usta and Korkmaz (2010) revealed that the main reason for students' failure in programming is not using correct teaching methods. In this respect, gamified activities, especially non-computer game activities, in pre-school and primary school years that develop algorithmic thinking skills which will form the basis of programming can be more beneficial (Aydoğdu, 2019). Thus, it is thought that the problem of children becoming addicted to computer as a passive consumer and the behavior of avoiding the computer by getting bored with the complex structure of programming may be prevented (Bell et al., 2009).

Many studies in the literature have found that activity-based practices without the use of computers are more beneficial in the development of the coding skills of students (Csernoch et al., 2015; Wohl et al., 2015). Atmatzidou and Demetriadis (2016) recommend game-based activities in which robotic kits are used instead of computers in the acquisition of coding skills. Figueiredo et al. (2021a) emphasized that algorithmic thinking should be able to think in cooperation with all curriculum areas, with all kinds of pedagogical strategies, game-based, in different situations and with different materials, even without any materials. In order to lay the foundation of computational thinking skills and support them in a computer-free environment, children should be supported in four basic skills, which are algorithmic tasks (for example, activities supporting visual perception and puzzle activities), logic-reasoning activities (for example, riddles, puzzles), the ability to follow directions (for example, mind games, strategy games), and analytical activities (for example, activities that support attention and problem solving skills) (Burton, 2010). It is thought that with these activities, children, for whom the foundations of algorithmic thinking skills are laid, will approach the problem situations they encounter in all areas of life more accurately and professionally, and the basis of programming will also be laid.

There is a consensus in the literature that computational thinking skills should be taught in early grades (Barr & Stephenson, 2011; Grover & Pea, 2013; Guzdial, 2008; Qualls & Sherrell, 2010; Yadav et al., 2017). Today, computational thinking skills are seen as a universal competence that every child should have (Bower et al., 2017), and they can be developed, which is also effective in adopting this view (Ioannou & Angeli, 2016; Hsu & Wang, 2018; Mezak & Papak, 2018; Tsalapatas et al., 2012). Within the framework of this requirement, the number of international projects carried out at the basic education and preschool education level is increasing day by day (Figueiredo et al., 2021b; Hoić-Božić et al., 2018; Mezak et al., 2021;).

However, the ways to integrate computational thinking skills into curricula, the current status of the pedagogical competencies of teachers in teaching computational skills, and the way to improving these competencies are discussed (Bower et al., 2017; Milkova, 2015). A current topic that needs to be investigated is the ways to develop computational thinking skills and the teaching of these skills in teacher education programs (Gretter & Yadav, 2016; Prieto-Rodriguez & Berretta, 2014; Yadav, et al., 2014). Our literature review has shown that the interventions to improve the computational thinking skills of teachers and prospective teachers have led to an improvement in participants in terms of the characteristics measured within the scope of the studies (Barr & Stephenson, 2011; Blum & Cortina, 2007; Bower, et al., 2017; Jaipal-Jamani & Angeli, 2017; Mouza et al., 2017; Prieto-Rodriguez & Berretta, 2014; Qian, et al., 2017; Yadav et al., 2014). This is promising for intervention research to be carried out on the development of teachers' computational thinking competencies.

In this study, based on the needs and gaps stated in the literature, activity-based teaching practices that support computational thinking skills were designed and implemented in order to improve the computational thinking and teaching skills of basic education teachers. The aim was to increase teachers' computational thinking skills and teaching competencies. The following questions were addressed in this study:

- Do activity-based computational thinking educational practices significantly contribute to teachers' computational thinking skills?
- How are the teachers' understandings of computational thinking skills?
- How are the teachers' understandings of the functions of computational thinking skills?
- How are the teachers' understandings of the principles of teaching computational thinking skills?
- How are the expectations and evaluations of the developers and practitioners of the activity-based computational thinking educational practices?

Method

Research Design

Based on the aim of the study, the case study design, one of the qualitative research designs, was selected in order to examine the experiences of the stakeholders (teachers, researchers, and trainers) in the process of computational thinking educational practices. Since a single case and a single analysis unit were studied in the research, the holistic single case design was adopted (Yin, 2014). Case study is an empirical research type that seeks answers to questions that begin with how and why specific to a current situation examined and provides the researcher with the opportunity to collect in-depth data (Yin, 2014). The case to be chosen as the research topic can be a person, student, administrator, program or class, school, organization, group, or anything of interest (Rabson, 2017). The case in this research is the activity-based computational thinking educational practices. The change that the participants showed in terms of computational thinking within the computational thinking educational practices and their understanding of computational thinking skills and teaching were examined in depth. In addition, the observations and evaluations of the developers and practitioners of computational thinking educational practices based on a total of 40 hours of training experiences lasting ten days were also examined. During the research period, data were obtained from three different sources using five data collection tools. According to Creswell (2013), case study is a qualitative research approach in which the researcher investigates one or more cases over some time with data collection tools (observations, interviews, audio-visuals, documents, reports) including multiple sources, and defines cases and case-related themes. In this research, the scope and effects of the activity-based computational thinking educational practices, which constitute the limited case, were investigated on the basis of the data obtained from multiple sources, and the case and the themes related to the case were examined in depth. The research process is as follows.

Table 1. Research Process

Examining the Effectiveness of Practices
Examination of the data obtained from the Computational Thinking Skills Scale before and after the practice
Examining the Concepts Regarding the Phenomena Constituting the Aim of the Practices
An in-depth examination of participants' understanding of computational thinking skills and teaching
Evaluations Based on Observations and Experiences Regarding the Implementation Process
Examination of the observations and evaluations based on the experiences of the developers and practitioners

Participants

The participants were selected using the criterion sampling method, one of the purposive sampling methods. Purposive sampling allows for the in-depth study of cases that are thought to have the potential to yield rich data on answering the research problem (Patton, 2014). The criterion sampling method is based on the idea of working with a group that meets all the predetermined set of criteria. The criterion for selecting the participants to be included in this study was to be a classroom teacher or a preschool teacher. The reason for this criterion is that the activity based computational thinking educational practices were developed for students at the basic education level. Other criteria were years of experience, city where the teacher works, and the educational status. The teachers who responded to the call for teacher participants all over Turkey were subjected to a selection process on the basis of the criteria specified above in a way that would ensure the greatest variety and impact. The characteristics of the participants are summarized in Table 2.

Table 2. Participants

Number/Name	Field	Gender	Years of experience	City of Türkiye	Educational status
1. A.Y.	CT	Male	5	Ağrı	Masters with thesis
2. A.K.	CT	Male	10	Ordu	Masters with thesis
3. A.E.	CT	Male	11	Manisa	Bachelor's degree
4. A.Ç.	CT	Male	13	Manisa	Bachelor's degree
5. K.Ç.	CT	Male	15	Samsun	Masters without thesis
6. M.Ö.	CT	Male	1	Bingöl	Bachelor's degree
7. Ö.N.	CT	Male	23	Eskişehir	Masters with thesis
8. R.D.	CT	Male	14	Afyon	PhD
9. D.B.	CT	Female	2	Batman	Bachelor's degree
10. D.A.	CT	Female	1	Ordu	Masters with thesis
11. E.G.	CT	Female	11	Kırşehir	Bachelor's degree
12. G.Y.	CT	Female	1	Çorum	Bachelor's degree
13. G.A.	CT	Female	1	Mersin	Bachelor's degree
14. M.T.	CT	Female	1	Bingöl	Bachelor's degree
15. P.G.	CT	Female	4	Muş	Bachelor's degree
16. R.K.	CT	Female	10	Ordu	Masters with thesis
17. S.O.	CT	Female	11	Adıyaman	Masters with thesis
18. S.K.	CT	Female	13	Mersin	PhD
19. Z.B.	CT	Female	1	Batman	Bachelor's degree
20. Z.N.	CT	Female	1	Bingöl	Bachelor's degree
21. A.A.	PT	Female	22	Ordu	Bachelor's degree
22. A.Y.	PT	Female	7	Rize	Bachelor's degree
23. B.Ç.	PT	Female	1	Şanlıurfa	Bachelor's degree
24. B.Ö.	PT	Female	10	Rize	Bachelor's degree
25. B.T.	PT	Female	7	Şanlıurfa	Bachelor's degree
26. B.G.	PT	Female	5	Diyarbakır	Bachelor's degree
27. E.G.	PT	Female	9	Aydın	Bachelor's degree
28. F.Y.	PT	Female	2	Gaziantep	Bachelor's degree
29. F.S.	PT	Female	2	Giresun	Masters with thesis
30. F.Z.	PT	Female	17	Adana	Masters with thesis
31. F.A.	PT	Female	11	İstanbul	Masters with thesis
32. F.Ö.	PT	Female	1	Zonguldak	Masters with thesis
33. F.K.	PT	Female	12	Adıyaman	Bachelor's degree
34. G.R.	PT	Female	12	İstanbul	Bachelor's degree
35. H.T.	PT	Female	1	Muğla	Bachelor's degree
36. H.A.	PT	Female	13	İstanbul	Bachelor's degree
37. H.K.	PT	Female	3	Diyarbakır	Bachelor's degree
38. N.T.	PT	Female	18	İstanbul	Bachelor's degree
39. S.Ç.	PT	Female	11	İstanbul	Bachelor's degree
40. Z.K.	PT	Female	8	Kocaeli	Bachelor's degree

Note: CT= Classroom teacher, PT=Preschool teacher

As seen in Table 2, a total of 40 teachers from all parts of Turkey participated in the study. Eight participants are male. The participants have different levels of education. They have work experience ranging from 1 to 24 years.

Data Collection Tools

Case study requires an in-depth questioning of how individuals see themselves based on their experiences, their perceptions according to the context, their emotions and the underlying reasons (Akar, 2016). In order to reveal perceptions and emotions, it is necessary to use different data sources and various data collection tools. In the data collection process, five data collection tools that serve the aims of the research were used. Table 3 summarizes the data collection tools used in the research.

Table 3. Data Collection Tools

Aim of the Tool	Data Collection Tool	Number of Participants
Examination of participants' level of computational thinking skills before and after the practice	Computational Thinking Skills Scale	40 teachers
An in-depth analysis of participants' understanding of computational thinking skills and teaching	Opinion form	40 teachers
An in-depth examination of the observations and inferences based on the experience of computational thinking skills educational practices.	Opinion form	4 researchers
	Field notes	2 researchers
	Opinion form	4 trainers

Computational Thinking Skills Scale

The scale was developed by Yağcı (2018) and consists of 42 items under four factors. It was used to measure the computational thinking skills of teachers in this research. The scale consists of the factors of Problem Solving (PS), Collaborative Learning and Critical Thinking (CL and CT), Diverse Thinking (DT), and Algorithmic Thinking (AT). The factor of Diverse Thinking includes items to evaluate the ability to think outside of the box and use imagination. The factor of Algorithmic Thinking includes items to evaluate the ability of individuals to solve any problem they encounter step by step by analyzing the problem and thinking and designing like a computer. The factor of Collaborative Learning and Critical Thinking includes items to evaluate teamwork skills and the skill of having an inquisitive approach to ideas. The factor of Problem Solving is the general characteristic of an individual's problem-solving skills. The Cronbach's alpha for the factors of the scale are as follows: .96 for PS, .94 for CL and CT, .94 for DT and .83 for AT. The Cronbach's Alpha for the whole scale was calculated as .97. Since the validity and reliability studies of the scale were not conducted on teachers, the Cronbach's alpha coefficients of the scale were recalculated within the scope of this study: .75 for PS, .72 for CL and CT, .76 for DT and .71 for AT. The items in the scale are scored on a five-point Likert-type scale rated from (1) strongly disagree to (5) completely agree.

Teacher opinion form

The form consisting of open-ended questions aims to examine the teachers' understanding of what computational thinking is, of its functions, and how the teaching process should be carried out. The questions in the form allow teachers to define algorithmic thinking skill at a conceptual level, to examine the functions of this thinking skill from a broad perspective, and to express their understanding of how this skill should be performed in the planning and implementation phases of the teaching process.

Trainer opinion form

The form consists of open-ended questions that allow teachers, who were trainers for 10 days for activity-based computational thinking educational practices, to share their experiences, contributions to the participants, and their inferences about what needs to be done in order to make the practices better.

Researcher opinion form

The form consists of open-ended questions that allow researchers, who are the developers of activity-based computational thinking educational practices and who make observations and interviews during the implementation process, to share their conclusions about the quality of the practices, their contributions to the participants, and what needs to be done to make the practices better.

Researcher field notes

The notes include evaluations based on the observations of two researchers, who are the developers of activity-based computational thinking educational practices, throughout the program.

Data Analysis

Analysis of the Quantitative Data

Whether the quantitative data were normally distributed was tested calculating the skewness and kurtosis coefficients. It was found that the data showed normal distribution. Thus, the Wilcoxon signed rank test was performed.

Analysis of the Qualitative Data

The qualitative data obtained from four different data collection tools serving two different aims were analyzed. The first aim was to examine the participant teachers' understanding of computational thinking skills and teaching in depth. In this context, the data obtained from the teacher opinion form was analyzed. The second aim was to analyze the experiences and observations of researchers and trainers regarding the computational thinking skills educational practices. Thus, the data obtained from the researcher field notes and trainer and researcher opinion forms were analyzed. The analysis process described in detail below was followed on both data sets.

Stage 1: Open coding. In the first stage of the analysis, open coding was performed on both data sets. The entire data set was thoroughly read. The data sets were divided into two as sentences or sentence groups and the research questions. In this process, 557 statements were identified in the first data set and 85 statements were identified in the second data set. In the open coding process, each expression was coded using the vivo code. Vivo codes are the codes created using words expressed by the interviewees (Strauss & Corbin, 1997). They are the words or phrases that best match the meaning of participants' statements. Choosing the names of the codes from among the statements of participants prevents possible mistakes that may arise from the researcher in the coding process. In this way, the influence of the researcher is minimized. All the analysis in the research was carried out based on these codes. Following this process, a total of 557 statements from 304 classroom and 153 preschool teachers in the first data set were coded using vivo code. In the second dataset, a total of 85 expressions from researchers and trainers were coded using vivo codes.

Stage 2: Classifying the dataset on the basis of the stages of the model. The process performed at this stage is axis coding. Axis coding is the process of associating the themes with the codes, and is the classification process of the data, which is broken down to the smallest detail with open coding, under categories. The purpose of axis coding is to classify the similar codes (Yıldırım & Şimşek, 2011). At this stage, the coded belief statements were examined in detail by posing the question of "What is this statement about?". In this process, the statements were re-read and classified on the basis of the issues determined to be relevant. At this stage, each of the coded expressions was classified according to the subject they are related to. The themes and the codes under these themes are presented in Table 4.

Table 4. Distribution of codes after axis coding

Stages	f	%
First Data Set		
What is computational thinking skill?	230	41.3
What are the functions of computational thinking skill?	182	32.7
How should computational thinking skills be taught?	145	26.0
Second Data Set		
Expectations from activity-based computational thinking educational practices	37	43.5
Evaluation of activity-based computational thinking educational practices	48	56.5

Stage 3: Classification of the data set under the factors. In order to answer the research questions, the statements classified on the basis of the relevant topics were categorized based on the perspectives they reflected. This stage, in which the continuous comparison analysis method was used, is defined by Glaser (1965) as the comparison of the data with other data coded under the same category. Analysis at this stage continued until the subcategories were unchanged. At this stage of the analysis, the expressions were classified under clusters on the basis of the points of view they reflect, using letters and without giving any names. The clusters were constantly compared with each other, and the necessary merging and demerging were performed. After reading the clusters several times, the classification process was completed, and the categories were given a name using the word or phrases that were thought to best reflect the classification sets in scope. This analysis revealed 18 subcategories in the first category, 14 in the second category, and 8 in the third category, specific to the first data set. In the second data set, 4 subcategories were reached in the first category and 2 subcategories were revealed in the second category.

Reliability and Ethics

Construct validity is a requirement in case studies. It requires defining the concepts under investigation with correct criteria. For this, it is important to use triangulation in the research process (Akar, 2016). In this research, methodological, data source and researcher triangulation were ensured. Methodological triangulation is concerned with obtaining similar or overlapping findings with different data collection techniques. Five different data collection tools were used in the study. In the data analysis process, the analysis steps for each tool were described in detail. As a part of data source triangulation, all stakeholders (teachers, trainers and researchers) were included in the research. As for researcher triangulation, the research was carried out by five faculty members. To ensure the reliability of the research, the number and characteristics of the participants, how they were selected, the data collection tools, and analysis techniques used in the research are expected to be explained in detail under the methodology part of the research (Creswell & Miller, 2000). It is recommended to use detailed description and sample selection strategies to fulfill the transferability requirement of the research (Merriam, 2009). The method part of the research was organized taking these requirements into account. The criterion sampling method was used in the study. In this way, the study was carried out with a group from which rich data could be obtained to answer the research questions. The participants are 40 teachers from all regions of Turkey, whose years of experience vary between 1-24 years. In addition, data was obtained from four trainers and the researchers. Transferability is also related to the fulfillment of analytical generalization conditions (Akar, 2016). In this context, the limitations of the research should also be stated. Transferability also involves making explanations about how research results can be transferred to people, units, places, or events. In this study, detailed information was given under the title of limitations of the research. To ensure the internal reliability of the research, opinions from different researchers were sought. Literature review was carried out to develop data collection tools, and the tools used for similar purposes in the literature were examined in terms of content and structure. After the development of the data collection tools, the opinions of three experts were received. In the study, research data were coded independently by two researchers to ensure intercoder reliability. The codes of two researchers were compared, and it was observed that there was a 97% agreement between the coders. Necessary changes were made until full consensus was reached.

Activity Based Computational Thinking Educational Practices

The main purpose of activity-based computational thinking educational practices is to improve the computational thinking skills of classroom teachers and preschool teachers and the competence for teaching these skills. For this reason, activity-based computational thinking educational practices were structured on two main competences, which are computational thinking skills and computational thinking teaching skills. The educational practices aim to develop the computational thinking skill on the basis of the sub-components of the skill, namely, decomposition, abstraction and pattern recognition. The sub-competence of computational thinking teaching skill is pedagogical competence. Another important aspect of the practices is that the practices aiming to develop sub-competences are also structured on the basis of the thinking skills they aim to develop. These thinking skills are analytical thinking, distinguishing, decision making, problem solving, spatial thinking, designing, transferring, and creative thinking. Each practice within the scope of the research was developed based on the core competences, sub-competences, and the relevant thinking skills. Activity-based computational thinking educational practices are summarized in Table 5.

Table 5. Activity Based Computational Thinking Educational Practices

Core competence	Sub-competence	Relevant thinking skills
Computational thinking skill	Decomposition	Analytical thinking (Analysis, classification, sequencing)
	Abstraction	Distinguishing Decision making
	Pattern recognition	Problem solving Spatial thinking
Computational thinking teaching skill	Pedagogical competence	Designing Transferring Creative thinking

Activity Based Computational Thinking Educational Practices is the product of the joint work of two classroom teaching experts, a preschool education expert, one instructional technologies expert, and one curriculum development expert. Prior to the design of the educational practices, firstly, a literature review was conducted. The theoretical components of computational thinking skills and the needs pointed out in the studies were identified. Then, games and practices that can improve computational thinking skills at the basic education level were examined. 20 different games and activities were included in the design of the practices, which were developed based on the core competences, sub-competences, and the relevant thinking skills, and attention was paid to the homogeneous distribution of the practices in the program. The distribution of activity-based computational thinking educational practices based on sub-competences and thinking skills is shown below.

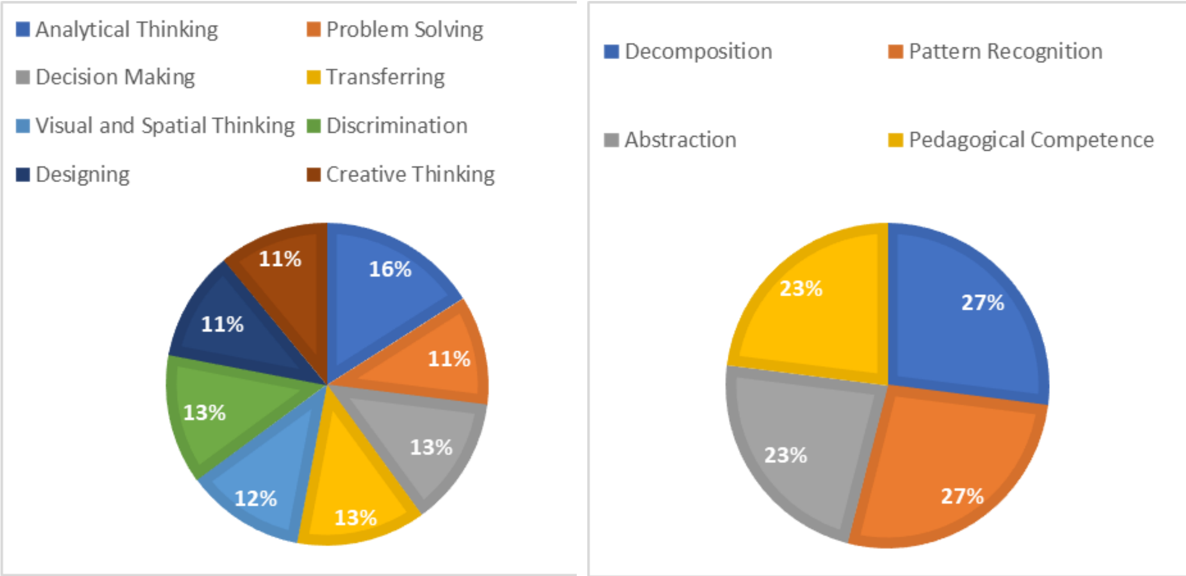

Figure 1. The Distribution of Activity-Based Computational Thinking Educational Practices on The Basis of Sub-Competences and Thinking Skills

All of the 20 activities developed in accordance with the aim of the program are mainly composed of games, drama and orff activities that ensure active participation. In each practice, the focus was on the development of the computational skills by the teachers as well as how the skill is taught. Table 6 shows the distribution of the subjects across educational practices.

Table 6. Activity-Based Computational Thinking Educational Practices across Subjects

Subject	Class hour	Thinking skill	Sub-competence
Patterns and algorithm	2	Analytical Thinking Spatial Thinking	Decomposition Pattern Recognition
Projection on the basis of computational thinking	1	Analytical Thinking Problem Solving Spatial Thinking	Decomposition Pattern Recognition
Mind games that develop computational thinking skills	1	Analytical Thinking Distinguishing Decision Making Problem Solving Spatial Thinking	Decomposition Abstraction Pattern Recognition
Using mind games that develop computational thinking skills as a teaching material	2	Analytical Thinking Distinguishing Decision Making Problem Solving Spatial Thinking Designing Transferring Creative Thinking	Decomposition Abstraction Pattern Recognition Pedagogical competence
Designing mind games that develop computational thinking skills as a teaching material	2	Analytical Thinking Distinguishing Decision Making	Decomposition Abstraction Pattern Recognition Pedagogical competence

		Problem Solving Spatial Thinking Designing Transferring Creative Thinking	
Designing algorithmic models from alternative materials	1	Designing Transferring Creative Thinking	Pedagogical competence
What I can do in my daily life with computational thinking Coding with pen and paper	2 1	Distinguishing Decision Making Analytical Thinking Spatial Thinking	Abstraction Decomposition Pattern Recognition
Music, movement, rhythm and algorithm	1	Analytical Thinking Distinguishing Decision Making Problem Solving	Decomposition Abstraction Pattern Recognition
Teaching algorithm with music, movement and rhythm	3	Designing Transferring Creative Thinking	Pedagogical competence
Designing educational games based on computational thinking	1	Designing Transferring Creative Thinking	Pedagogical competence
Using computational thinking based mobile applications as a teaching material	1	Analytical Thinking Distinguishing Decision Making	Decomposition Abstraction
Computational thinking based educational games	1	Analytical Thinking Distinguishing Decision Making Problem Solving Spatial Thinking Designing Transferring Creative Thinking	Decomposition Abstraction Pattern Recognition Pedagogical competence
Using coding educational robots as a teaching material	2	Analytical Thinking Distinguishing Decision Making Problem Solving Spatial Thinking Designing Transferring Creative Thinking	Decomposition Abstraction Pattern Recognition Pedagogical competence

Findings

The Wilcoxon signed rank test results regarding the effect of activity-based computational thinking educational practices on preschool teachers' computational thinking skills are summarized in Table 7.

Table 7. The Effect of Activity-Based Computational Thinking Educational Practices on Preschool Teachers' Computational Thinking Skills (N=20)

Factors		X	SD	Z	P
Problem Solving	Pre-test	84.55	7.09	-1.852	0.064
	Post-test	86.75	7.79		
Collaborative Learning /Critical Thinking	Pre-test	32.9	3.81	-0.327	0.743
	Post-test	33.3	4.15		
Diverse Thinking	Pre-test	36.5	3.28	-1.773	0.076
	Post-test	39.2	3.84		
Algorithmic Thinking	Pre-test	18.85	2.15	-3.535	0.000**
	Post-test	22.0	2.32		
Computational Thinking Total Score	Pre-test	172.80	11.87	2.373	0.018*
	Post-test	184.25	11.72		

**p<0.05; **p<0.005*

When Table 7 is examined, it is seen that as a result of the educational activities of the preschool teachers, among the computational thinking skills, there is a statistically significant increase only in algorithmic thinking skills (z=-3.535; p<0.005) and computational thinking skills total score (z=2.373; p<0.005). It was found that there was no significant difference in problem solving, cooperative learning/critical thinking and diverse thinking skills after the education. In this context, it can be said that activity-based computational thinking education contributed significantly to the algorithmic thinking skills and total computational thinking scores of the preschool teachers. The Wilcoxon signed rank test results regarding the effect of activity-based computational thinking educational practices on classroom teachers' computational thinking skills are summarized in Table 8.

Table 8. The Effect of Activity-Based Computational Thinking Educational Practices on Classroom Teachers' Computational Thinking Skills (N=20)

Factors		X	SD	Z	P
Problem Solving	Pre-test	81.25	5.57	-3.605	0.000**
	Post-test	91.5	8.33		
Collaborative Learning /Critical Thinking	Pre-test	33.85	3.95	-1.169	0.242
	Post-test	35.2	4.03		
Diverse Thinking	Pre-test	36.5	3.59	-2.172	0.030*
	Post-test	38.95	4.35		
Algorithmic Thinking	Pre-test	19.4	2.28	-2.967	0.003**
	Post-test	21.65	2.81		
Computational Thinking Total Score	Pre-test	171.00	11.24	3.333	0.001**
	Post-test	187.30	14.95		

**p<0.05; **p<0.005*

When Table 8 is examined, it is seen that classroom teachers' problem solving (z=-3.605; p<0.005), diverse thinking (z=-2.172; p<0.05), and algorithmic thinking skills (z=-2.967; p<0.005) and total computational thinking scores (z=3.333; p<0.005) increased statistically significantly. On the other hand, there was no significant difference in collaborative learning/critical thinking skills after the education. In this context, it can be said that activity-based computational thinking education contributed to classroom teachers' problem solving, diverse thinking, and algorithmic thinking skills, and computational thinking total scores. The concept map for 230 statements reflecting the teachers' understanding of computational thinking skill is presented in Figure 2.

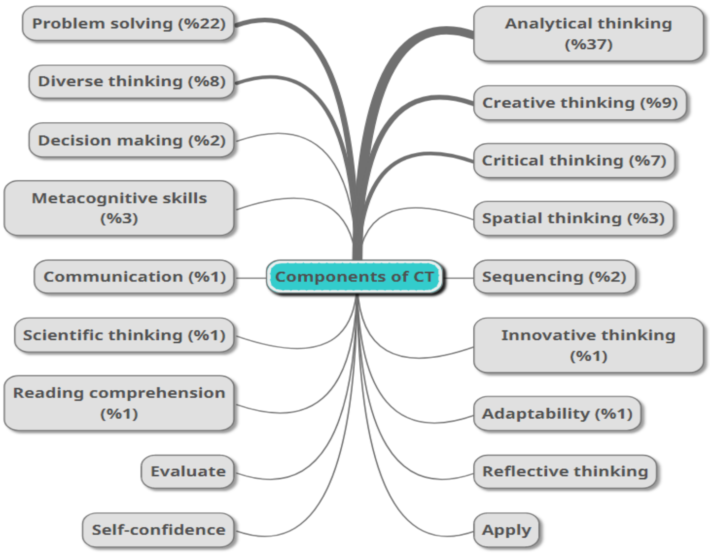

Figure 2. Definition of computational thinking skills

When the concept map for the 230 expressions reflecting teachers' understanding of computational thinking skills is examined, it is seen that these concepts can be grouped under 18 different thinking skills. In other words, it is observed that teachers defined computational thinking with 18 different thinking skills. Among these thinking skills, the most frequently emphasized skills are analytical thinking, problem solving, creative thinking, multidimensional thinking and critical thinking. These thinking skills are followed by metacognitive skills, spatial thinking, sequencing and communication skills. Self-confidence, reflective thinking, evaluation and application skills are among the skills that are rarely expressed. Accordingly, it can be said that teachers defined computational thinking as analytical thinking, problem solving, creative thinking, critical thinking and multidimensional thinking skills. The concept map for 182 expressions reflecting teachers' understanding of the functions of computational thinking is presented in Figure 3.

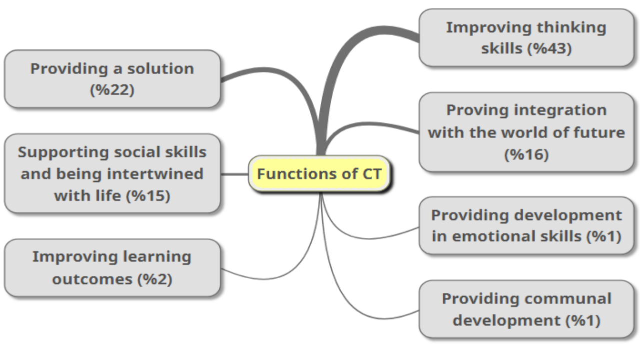

Figure 3. Functions of Computational Thinking Skills

When the concept map for 182 expressions reflecting teachers' understanding of the functions of computational thinking is examined, it is seen that the expressions were grouped under seven different functions. In other words, it is seen that the teachers defined the functions of computational thinking under seven main headings. The function with the highest rate is improving thinking skills, which is followed by the function of providing solutions to problems.

The teachers stated that computational thinking is the newest, shortest, easiest, fastest, most valid, reliable, and economical way to solving all kinds of problems. Another prominent function of computational thinking stated by the participants is the importance of computational thinking in integrating into the world of the future. The relationship of computational thinking with the 21st century skills and technology are among the points emphasized in this context. Another function stated by the participants is that computational thinking is intertwined with life. In this context, it is stated that the awareness and inspiration created by encountering computational thinking in social and human life plays a role in involving the individual in daily life more effectively. It has also been stated that computational thinking skills create an improvement in teacher qualifications and in the social and emotional aspects of the individual. The concept map for 145 expressions related to teachers' understanding of the principles of teaching computational thinking is presented in Figure 4.

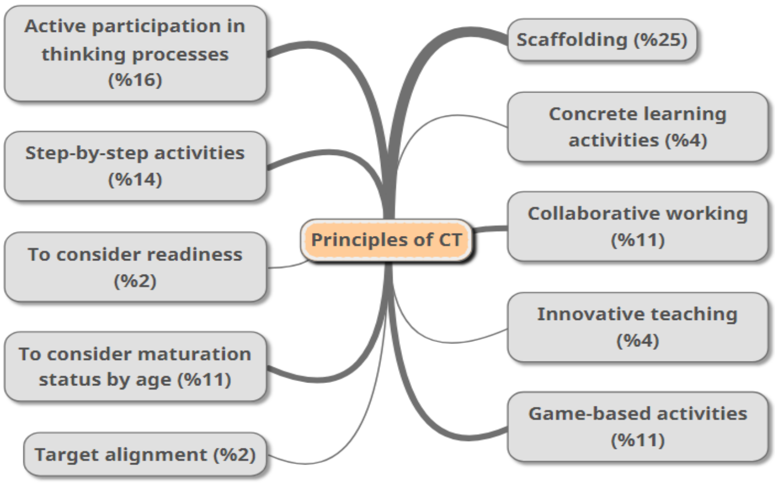

Figure 4. Participants' Understanding of the Principles of Teaching Computational Thinking

When the concept map for 145 expressions related to teachers' understanding of the principles of teaching computational thinking is examined, it is seen that the expressions can be grouped under 11 principles. In other words, the teachers defined 11 principles for teaching computational thinking. When the principles that should be considered in the teaching of computational thinking skills are examined, it is seen that the teachers mostly pointed to the importance of using scaffolding. In addition, the teachers stated that students' active participation in thinking processes should be ensured; students should work in cooperation; game-based activities should be used, and innovative teaching practices should be included in the program. The steps of the teaching process proposed by the teachers were categorized with the analysis carried out on the instructional designs of computational thinking skills of the participants who completed the educational practices. The 14-step teaching process indicated by the teachers' statements is shown in Figure 5.

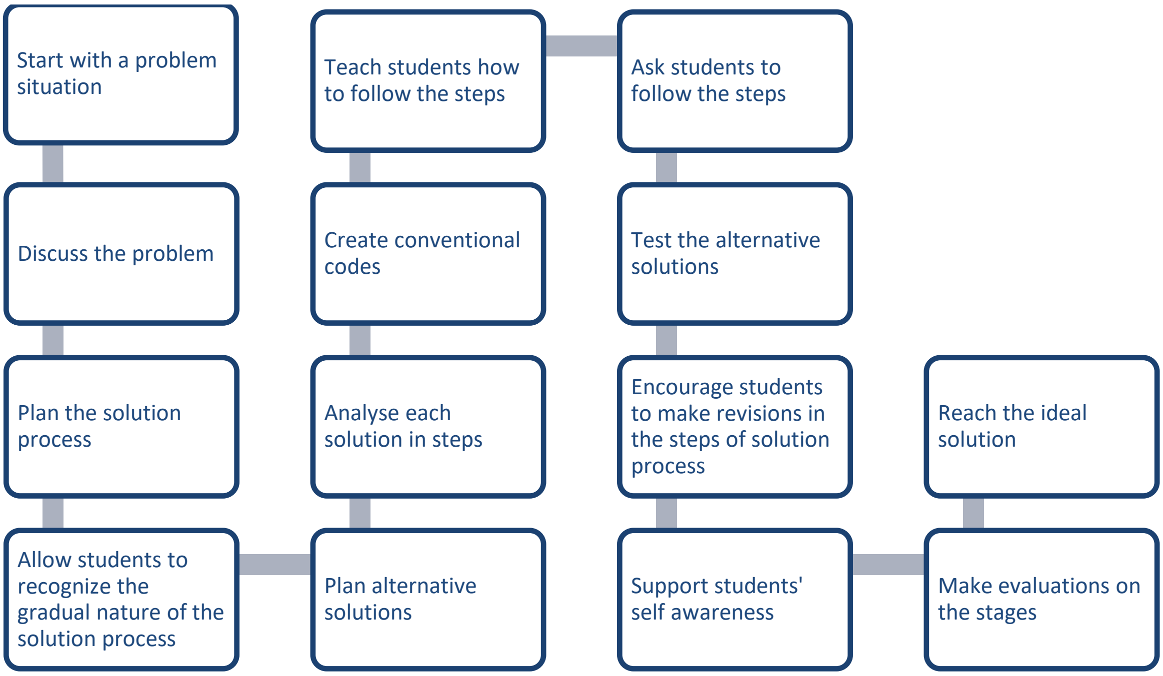

Figure 5. Teaching Steps of Computational Thinking according to the Participants

The teachers pointed out that the computational thinking teaching process should be carried out by following the steps in Figure 5. Every step described here is directed by the teacher and experienced with the students. The steps are as follows: Start with a problem situation, discuss the problem situation, plan the solution process of the problem, carry out studies that allow students to feel the gradual (consisting of a series of steps) nature of the solution process, plan all alternative ways to solve the problem, analyze each solution in steps, create conventional codes within the group on solution ways in order not to experience communication barriers in the process, teach students how to perform the solution steps, ask students to follow the steps, enable students to choose, experiment and evaluate alternative solutions, encourage students to make necessary revisions in the steps of the solution process, share tools that will enable students to develop self-awareness within the process they experience, enable students to make evaluations on the basis of stages at the end of the process (make an evaluation on the process, not the result, ensure that those who encounter erroneous results do not see the whole process as faulty, determine where the error originates from with an analytical understanding), and reach the ideal solution. The concept map for 83 statements of four researchers and four trainers, determined based on the expectations and evaluations of the developers and practitioners of computational thinking teaching practices, is presented in Figure 6.

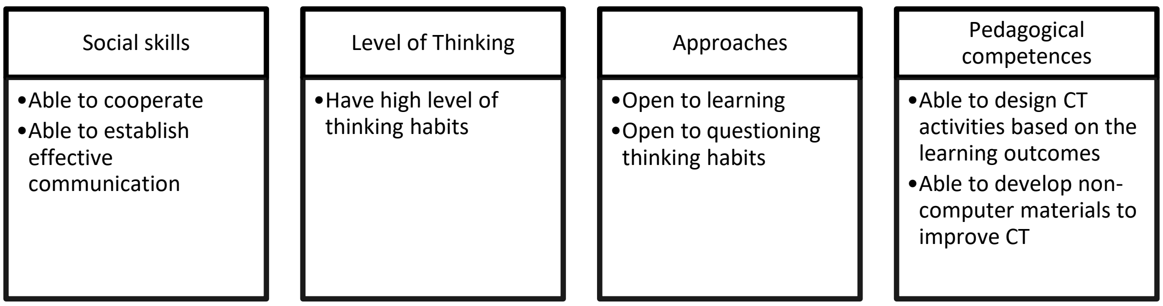

Figure 6. Expectations of Developers and Practitioners of Computational Thinking Teaching Practices

Figure 6 presents the expectations of the developers and practitioners of computational thinking teaching practices under four categories. These are expectations for social skills, thinking levels, approaches, and pedagogical competencies. Expectations regarding social skills were discussed under the headings of being able to cooperate and communicate effectively. Expectations for level of thinking are having high-level thinking habits. Expectations for approaches are being open to learning and being open to questioning thinking habits. Expectations regarding pedagogical competencies are designing CT activities suitable for learning outcomes and developing non-computer materials to develop computational thinking skills. The evaluations of the developers and practitioners of computational thinking teaching practices based on their observations and experiences during the implementation process are presented in Figure 7.

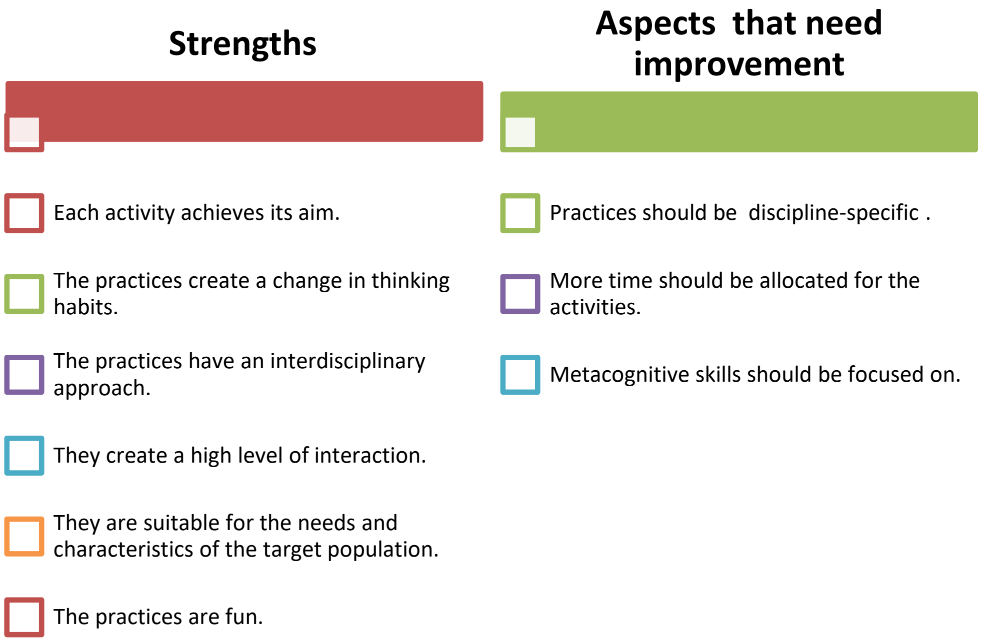

Figure 7. Evaluation of the Practices

Figure 7 presents the evaluations of the developers and trainers under two categories as strengths and aspects that need improvement. The fact that the practices achieve their aims, create a change in the thinking habits of the participants, have an interdisciplinary approach, and create a high level of interaction are among the strengths of the practices. The most recurring aspect that needs to be improved in the program is the need to increase the time allocated to the activities. In addition, it was stated that practices that develop the metacognitive thinking skills of the participants should be included, and the activities should be discipline-specific.

Discussion and Conclusion

This study aimed to carry out an in-depth evaluation of the activity-based computational thinking teaching practices performed to improve the computational thinking and teaching skills of basic education teachers. The findings of the research suggest that the educational practices led to a statistically significant increase in the problem solving, diverse thinking and algorithmic thinking skills of the classroom teachers, and in the algorithmic thinking skills of the preschool teachers. It was found that the participants defined computational thinking with 18 different thinking skills. They stated that the most fundamental function of computational thinking skills is to improve students' thinking skills. The second most recurring function of computational thinking is that it provides solutions to problems. Another prominent function stated by the participants is the importance of computational thinking skills in integrating into the world of the future. The relationship of computational thinking skills with the 21st century skills and technology are among the points emphasized in this context. It has also been stated that computational thinking skills lead to an improvement in teacher qualifications and in the social and emotional aspects of the individual. When the principles that should be considered in the teaching of computational thinking skills were examined, it was seen that according to the teachers, scaffolding should be used most frequently. In addition, as a result of the analysis carried out on the instructional designs of the participants for the teaching of algorithmic thinking skills, a 14-step teaching path explained in detail in the findings section was described. It is thought that this teaching path can be a guide for intervention studies to be carried out to improve the teaching of computational thinking in the future. The analysis based on the opinions of the stakeholders of the project shows that the practices carried out within the scope of the project are an effective tool for the development of teachers' computational thinking and teaching skills.

The study revealed that the participants defined computational thinking on the basis of 18 different thinking skills. Almost all of these thinking skills are 21st century skills. The literature also supports this finding of the research. Computational thinking is considered to be a fundamental component with a high potential to fulfil the requirements of the 21st century from the individual (Angeli & Valanides, 2020; Yadav et al., 2014; Yadav et al., 2017). In this respect, it can be said that this study brought the participants to a point of a common understanding with the literature. It was seen that the participants explained computational thinking mainly on the basis of analytical thinking, problem solving, critical thinking and multidimensional thinking skills. When the literature is examined, it is seen that computational thinking is defined as a high-level thinking skill (Yadav et al., 2014), in which problem-solving skills are used intensively. Yadav et al. (2017) stated that pre-service teachers defined computational thinking skills mainly on the basis of problem solving and logical thinking skills. Supporting the research findings, ISTE (2015) also states that computational thinking includes creativity, algorithmic thinking, critical thinking, problem solving, and communication and cooperation skills. In this context, studies conducted to develop measurement tools show that this skill has a statistically significant relationship with spatial thinking, reasoning and problem solving, supporting our research findings (Román-González et al., 2017). These explanations reveal the high potential of computational thinking skills in developing higher-order thinking skills. Computational thinking skill has a critical role in the development of other thinking skills. Research findings suggest that other thinking skills can be developed with the teaching of computational thinking skills, as highlighted in the literature.

The teaching of activity-based computational thinking contributed to classroom teachers' problem solving, diverse thinking and algorithmic thinking skills and to the algorithmic thinking skills of preschool teachers. This shows that activity-based computational thinking educational practices can be used to improve teachers' computational thinking skills. The qualitative data of the study also support this finding. The developers and practitioners of the activities also stated that the activities achieved their goals. The practices were found to change the thinking habits of the participants, create a high level of interaction, and be suitable for the needs and characteristics of the participants. In fact, the literature shows that interventions with different durations and characteristics aimed at improving the computational thinking skills of teachers and prospective teachers caused an improvement in the participants within the scope of the characteristics measured in the research (Barr & Stephenson, 2011; Blum & Cortina, 2007; Bower, et al., 2017; Jaipal-Jamani & Angeli, 2017; Mouza et al., 2017; Pala & Mıhçı-Türker, 2020; Prieto-Rodriguez & Berretta, 2014; Qian et al., 2018; Yadav et al., 2014; Yadav et al., 2017;).

It is known that how computational thinking can be developed in teacher education is a current and much needed topic (Gretter & Yadav, 2016; Tang et al., 2020). There is evidence that trainers seem to lack the necessary knowledge and skill to teach computational skills to prospective teachers (Angeli & Valanides, 2020; Bonani et al., 2022; Yadav et al., 2011; Yadav et al., 2014). In-service teacher education practices are not different at this point. Bonani et al. (2022) state that teachers are generally not familiar with computational thinking teaching practices. This research showed that teachers need computational thinking and teaching. The interviews with the participants at the beginning of the activities and the field notes on joint activities showed that teachers' perspectives on computational thinking and teaching are limited and superficial. However, in order for students to acquire this important skill, teachers need to have in-depth knowledge of teaching computational thinking (Hodhod et al., 2016). When the literature is examined, it is seen that some program development studies have been carried out on this subject (Angeli et al., 2016; Güler, 2021). In this context, it is important to underline the need to conduct teaching practice development studies, an example of which is this research.

One of the main purposes of the activities carried out within the scope of the project was to improve the computational thinking teaching skills. In this context, the study aimed to develop the understanding of teachers on what kind of principles should be taken into account in the teaching of computational thinking skills. The teachers in the study stated that scaffolding should be used most frequently in teaching computational thinking. It is also emphasized in the literature that scaffolding is one of the most effective ways to teach high-level thinking skills (Rosenshine & Meister, 1992). Futschek and Moschitz (2010) also explain the role of the teacher in the process of teaching computational thinking on the basis of scaffolding. They explain that in teaching computational thinking, it is necessary for the teacher to give appropriate problem statements and ask appropriate questions in order for the students to think of creating algorithms that lead to the solution of these problems. Guiding students with questions in the process to a solution is undoubtedly an integral part of the scaffolding process. Similarly, regarding how the teaching of computational thinking should be carried out, Mezak et al. (2021) stated that the problem should be understood, how it can be solved should be evaluated, the best strategies should be selected, and the problem should be divided into smaller sub-problems and solved step by step.

Regarding the principles of teaching computational thinking, the participants also stated that students' active participation in the thinking processes should be ensured, students should work cooperatively, game-based activities should be used, each step of activities should be explained, and innovative teaching practices should be performed. Doğan (2020) also pointed to the importance of using techniques such as active participation, learning by discovery, problem solving, induction, brainstorming, concept mapping, game, discussion, and case study. Highlighting the importance of active involvement of students in thinking processes, Mazak et al. (2021) defined the roles of teachers in teaching computational thinking skills as guiding students to understand, evaluate and recognize the problem and helping students take steps to solve the problem, whether correct or not. As a matter of fact, it is seen in the literature that in order to improve students' computational thinking skills, techniques such as learning scenarios (Bonani et al., 2022; Mezak & Papak, 2018), game-based programming (Tsalapatas et al., 2012) game mechanics (Hsu & Wang, 2018), multimedia (Milkova, 2015), and scaffolding (Angeli & Valanides, 2020) are used.

One of the outcomes of this research regarding the teaching of computational thinking is the classification of the steps of teaching computational thinking. The steps of the teaching process revealed based on the analysis of the computational thinking teaching designs requested from the teachers can be tested and developed within the scope of intervention studies to be conducted in the field. Although there are program and intervention studies on developing computational thinking skills in the literature, studies on how to teach computational thinking step by step are not common. It is stated in the literature that more studies should be conducted on this subject and that this field is in its infancy (Yadav et al., 2014). It is noteworthy that this teaching process was reached upon the detailed analysis of the instructional designs of the 40 teachers who participated in the project. It is believed that the 14-step teaching process that leads students to the most ideal solution by starting with a problem situation is a step towards eliminating the gap in the literature.

All the teachers trained in this study stated that computational thinking skills can be integrated into all courses. The teachers even recommended that this education be structured in the form of lesson-specific teaching practices in the future. Similar recommendations were made by Hill et al. (2004), Yadav et al. (2014) and Mezak et al. (2021). Emphasizing the importance of developing teachers' computational thinking in the context of the subject they teach, Hill et al. (2004), stated that unless teachers' knowledge is developed in the context of the content they will teach, their understanding in this context will be limited to an "abstract" understanding. Brown et al. (1989) argued that teachers cannot incorporate the inactive information into teaching practices as a result of this very situation. Therefore, while teaching computational thinking, not only pedagogy (how to teach) but also content knowledge (such as mathematics and literacy) and pedagogical content knowledge (for example, knowledge of mathematics for teaching) should also be considered (Hill et al. 2004). Yadav et al. (2014) stated that future studies should be based on the collaboration of trainers and computer scientists to develop concrete examples of how computational thinking can be embedded in key content areas from literacy and art to mathematics and science. In this context, the studies by Barr and Stephenson (2011), Weintrop et al. (2016), and Barr et al. (2011) are noteworthy. In our study, the teachers, who were convinced of the power of computational thinking to develop interdisciplinarity and thinking skills, recommended that the next teaching practice should be carried out in a more content-oriented approach, in a corresponding manner to what is recommended in the literature.

In this study, the focus was on developing computational thinking skills with activities other than computers. At the end of the education, all the participants were able to design effective teaching practices to develop computational thinking without using computers. The idea that teaching of computational thinking can only be performed using computers is also criticized by some experts in the field. Yadav et al. (2017) argued that computer is a context for the development of computational thinking, but it is not the only way to develop this skill. According to Mezak et al. (2021), computational thinking is not only about computer science and programming. Hemmendinger (2010) stated that teaching computational thinking is about teaching individuals how to use computational thinking to solve their problems and make productive discoveries in their thinking processes like an economist, a physicist, or an artist. Hemmendinger (2010) further emphasized that computational thinking is not thinking like a computer expert. Mezak et al. (2021) maintained that children can easily learn computational thinking by using examples from their daily lives, as it is the case in the activities developed for this research. Bonani et al. (2022) also posit that non-computerized interventions encourage computational thinking.

Based on the research findings, the following suggestions can be made:

- It can be recommended to improve the curricula of both teacher education and on-the-job education programs to improve the computational thinking skills of basic education teachers.
- Models on how teachers should teach their students computational thinking skills can be developed and studies can be conducted to test their effectiveness.
- It can be suggested that the model put forward within the scope of this research should be used to develop teachers' computational thinking skills.
- It can be suggested that more activity examples be developed to guide teachers in teaching computational thinking skills in computer-free environments.
- This study, which was carried out with preschool teachers and classroom teachers, can be taken as a model to be examined at different education levels.
- Courses on the development of algorithmic thinking skills can be included in teacher training programs implemented in education faculties.

Acknowledgement

This study was produced from the studies carried out in the "Activity-Based Algorithmic Thinking Education" project numbered 121B282 supported within the scope of Tübitak 4005 Science and Society projects. We thank Tübitak for their support and Ordu University and Amasya University for their contributions.

References

Akar, H. (2016). Case study. *Qualitative research patterns in education.* Ani Publishing.

Angeli, C. (2021). The effects of scaffolded programming scripts on pre-service teachers' computational thinking: Developing algorithmic thinking through programming robots. *International Journal of Child-Computer Interaction,* 100329. https://doi.org/10.1016/j.ijcci.2021.100329

Angeli, C., & Valanides, N. (2020). Developing young children's computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy. *Computers in Human Behavior*, *105*, 105954. https://doi.org/10.1016/j.chb.2019.03.018

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., & Zagami, J. (2016). A K-6 computational thinking curriculum framework: Implications for teacher knowledge. *Journal of Educational Technology & Society*, *19*(3), 47-57. Available at: http://www.jstor.org/stable/jeductechsoci.19.3.47 Access date: 12.12.2021.

Atmatzidou, S. & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems, 75,* 661-670. https://doi.org/10.1016/j.robot.2015.10.008

Aydoğdu, E. (2019). *Examination of students' algorithmic thinking skills in the process of non-computer activities*. [Unpublished Master thesis]. Trabzon University.

Barr, D., Harrison, J., & Conery, L. (2011). *Computational thinking: a digital age skill for everyone,* Available at: http://files.eric.ed.gov/fulltext/EJ918910.pdf Access date: 1.12.2021.

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

Bell, T., Alexander, J., Freeman, J. & 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. Available at: https://purehost.bath.ac.uk/ws/files/214932627/NZJACIT_Unplugged.pdf Access date: 20.11.2021.

Blum, L., & Cortina, T. J. (2007). CS4HS: An outreach program for high school CS teachers. *ACM SIGCSE Bulletin*, *39*(1), 19-23. https://doi.org/10.1145/1227504.1227320

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016, June). Exploring the field of computational thinking as a 21st century skill. In *Proceedings of the International Conference on Education and New Learning Technologies*, 4-6 July 2016, Barcelona, Spain (pp. 4725-4733).

Bonani, A., Gennari, R., & Mahlknecht, G. (2022). Scenarios for Graph Algorithmic Thinking Co-created with Teachers. In: De la Prieta F. et al. (eds) *Methodologies and Intelligent Systems for Technology Enhanced Learning, 11th International Conference. MIS4TEL 2021*. Lecture Notes in Networks and Systems, vol 326. Springer, Cham. https://doi.org/10.1007/978-3-030-86618-1_5

Bower, M., Wood, L. N., Lai, J. W., Highfield, K., Veal, J., Howe, C., ... & Mason, R. (2017). Improving the computational thinking pedagogical capabilities of school teachers. *Australian Journal of Teacher Education*, *42*(3), 53-72. https://doi.org/10.14221/ajte.2017v42n3.4

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

Burton, B. A. (2010). Encouraging algorithmic thinking without a computer. *Olympiads in Informatics, 4,* 3-14. Available at: https://ioinformatics.org/journal/INFOL053.pdf Access date: 1.12.2021.

Creswell, J. W. (2013). *Qualitative research methods: Qualitative research design according to five approaches* (M. Tüm & S. B. Demir, Trans.). Ankara: Political Publication Distribution.

Creswell, J. W., & Miller, D. L. (2000). Determining validity in qualitative inquiry. *Theory into Practice*, *39*(3), 124-130. https://doi.org/10.1207/s15430421tip3903_2

Csernoch, M., Biró, P., Máth, J. & Abari, K. (2015) Testing algorithmic skills in traditional and non-traditional programming environments. *Informatics in Education*, *14*, 175-197. Available at: https://www.ceeol.com/search/article-detail?id=342794 Access date: 12.12.2021.

Curzon, P. (2015). *Computational Thinking: Searching to Speak*. Available at: http://teachinglondoncomputing.org/free-workshops/computational-thinking-searching-to-speak/ Access date: 20.10.2020

Denning, P. J., & Tedre, M. (2021). Computational thinking for professionals. *Communications of the ACM*, *64*(12), 30-33. https://doi.org/10.1145/3491268

Doğan, A. (2020). Algorithmic thinking in primary education. *International Journal of Progressive Education, 16*(4), 286-301. https://doi.org/10.29329/ijpe.2020.268.18

Figueiredo, M., Gomes, C. A., Amante, S., Gomes, H., Alves, V., Duarte, R. P., & Rego, B. (2021a, September). Play, Algorithmic Thinking and Early Childhood Education: Challenges in the Portuguese Context. In 2021 International Symposium on Computers in Education (SIIE) (pp. 1-4). IEEE.

Figueiredo, M., Amante, S., Gomes, H. M. D. S. V., Gomes, M. A., Rego, B., Alves, V., & Duarte, R. P. (2021b). Algorithmic thinking in early childhood education: Opportunities and supports in the Portuguese context. In EDULEARN21 Proceedings (pp. 9339-9348). IATED.

Futschek, G., & Moschitz, J. (2010). Developing algorithmic thinking by inventing and playing algorithms. *Proceedings of the 2010 Constructionist Approaches to Creative Learning, Thinking and Education: Lessons for the 21st Century (Constructionism 2010)*, 1-10.

Glaser, B. G. (1965). The constant comparative method of qualitative analysis. *Social Problems*, *12*(4), 436-445. https://doi.org/10.2307/798843

Gomes, A. & Mendes, A. J. (2007, September). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE.* Available at: https://www.ineer.org/Events/ICEE2007/papers/411.pdf Access date: 12.12.2021.

Gretter, S., & Yadav, A. (2016). Computational thinking and media & information literacy: An integrated approach to teaching twenty-first century skills. *TechTrends, 60*(5), 510-516. https://doi.org/10.1007/s11528-016-0098-4

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

Guzdial, M. (2008). Education paving the way for computational thinking. *Communications of the ACM, 51*(8), 25-27. https://doi.org/10.1145/1378704.1378713

Güler, Ç. (2021). Algorithmic Thinking Skills without Computers for Prospective Computer Science Teachers. *Journal of Theoretical Educational Science, 14*(4), 570-585. https://doi.org/10.30831/akukeg.892869

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

Hill, H. C., Schilling, S. G., & Ball, D. L. (2004). Developing measures of teachers' mathematics knowledge for teaching. *The Elementary School Journal, 105*(1), 11-30. Available at: http://www.sii.soe.umich.edu/newsite_temp/documents/hill_schill_ball.db.r59E46.pdf Access date: 1.12.2021.

Hodhod, R., Khan, S., Kurt-Peker, Y., & Ray, L. (2016, February). Education teachers to integrate computational thinking into K-12 teaching. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education* (pp. 156-157). https://doi.org/10.1145/2839509.2844675

Hoić-Božić, N., Dlab, M. H., Prskalo, L. N., Rugelj, J., & Šerbec, I. N. (2018). Games for learning algorithmic thinking–GLAT Project. *International Journal of Multidisciplinary Research, 4*(2), 73-95. Available at: https://glat.uniri.hr/wp-content/uploads/2018/08/GLAT_ICEDL_2018_FINAL.pdf Access date: 3.12.2021.

Hoyles, C., & Noss, R. (2015). A computational lens on design research. *ZDM Mathematics Education 47*, 1039–1045. https://doi.org/10.1007/s11858-015-0731-2

Hsu, C. C., & Wang, T. I. (2018). Applying game mechanics and student-generated questions to an online puzzle-based game learning system to promote algorithmic thinking skills. *Computers & Education, 121*, 73-88. https://doi.org/10.1016/j.compedu.2018.02.002

Ioannou, I., & Angeli, C. (2016). A framework and an instructional design model for the development of students' computational and algorithmic thinking. In *MCIS* (p. 19). Available at: https://aisel.aisnet.org/mcis2016/19 Access date: 1.12.2021.

ISTE (2015). *CT Leadership toolkit*. Available at http://www.iste.org/docs/ct-documents/ct-leadershipt-toolkit.pdf?sfvrsn=4 Access date: 20.10.2020.

Jaipal-Jamani, K., & Angeli, C. (2017). Effect of robotics on elementary preservice teachers' self-efficacy, science learning, and computational thinking. *Journal of Science Education and Technology, 26*(2), 175-192. https://doi.org/10.1007/s10956-016-9663-z

Karabak, D., & Güneş, A. (2013). Curriculum proposal for first class secondary school students in the field of software development. *Journal of Research in Education and Teaching, 2*(3), 175-181. Available at: http://jret.org/FileUpload/ks281142/File/21.karabak.pdf Access date: 13.12.2021.

Korkmaz, Ö., Çakır, R., Özden, M.Y. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). *Computers in Human Behaviour, 72*, 558-569. https://doi.org/10.1016/j.chb.2017.01.005

Lahtinen, E., Ala-Mutka, K. & Järvinen, H. (2005). A study of the difficulties of novice programmers. *ACM SIGCSE Bulletin, 37*(3), 14-18. https://doi.org/10.1145/1151954.1067453

Merriam, S. (2009). *Qualitative research: A guide to design and implementation.* San Fransisco: John Willey & Sons Inc.

Mezak, J., & Papak, P. P. (2018, May). Learning scenarios and encouraging algorithmic thinking. *41st International Convention on Information and Communication Technology, Electronics* *and Microelectronics (MIPRO)*, 2018, pp. 760-765. https://doi.org/10.23919/MIPRO.2018.8400141

Mezak, J., Papak, P. P., & Vujičić, L. (2021, July). The integration of algorithmic thinking into preschool education. In *Proceedings of EDULEARN21 Conference*, 5-6 July 2021, (Vol. 5, p. 6).

Milkova, E. (2015). Multimedia application for educational purposes: Development of algorithmic thinking. *Applied Computing and Informatics*, *11*(1), 76-88. https://doi.org/10.1016/j.aci.2014.05.001

Mouza, C., Yang, H., Pan, Y. C., Ozden, S. Y., & Pollock, L. (2017). Resetting educational technology coursework for pre-service teachers: A computational thinking approach to the development of technological pedagogical content knowledge (TPACK). *Australasian Journal of Educational Technology*, *33*(3). 61-76. https://doi.org/10.14742/ajet.3521

Pala, F. K., & Mıhçı-Türker, P. (2020). The effects of different programming trainings on the computational thinking skills. *Interactive Learning Environments*, *29*(7), 1090-1100. https://doi.org/10.1080/10494820.2019.1635495

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

Park, H., Kim, H. S., & Park, H. W. (2020). A scientometric study of digital literacy, ICT literacy, information literacy, and media literacy. *Journal of Data and Information Science*, *6*(2), 116-138. https://doi.org/10.2478/jdis-2021-0001

Patton, M. Q. (2014). *Qualitative research & evaluation methods: Integrating theory and practice*. Sage publications.

Perry, G. (2009). *Programming guide for beginners* (T. Aksoy, Trans. ed.). Istanbul: Sistem Publishing.

Prieto-Rodriguez, E., & Berretta, R. (2014, October). Digital technology teachers' perceptions of computer science: It is not all about programming. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings* (pp. 1-5). https://doi.org/10.1109/FIE.2014.7044134

Qian, Y., Hambrusch, S., Yadav, A., & Gretter, S. (2018). Who needs what: Recommendations for designing effective online professional development for computer science teachers? *Journal of Research on Technology in Education*, *50*(2), 164-181. https://doi.org/10.1080/15391523.2018.1433565

Qualls, J. A., & Sherrell, L. B. (2010). Why computational thinking should be integrated into the curriculum. *Journal of Computing Sciences in Colleges*, *25*(5), 66-71.

Rabson, C. (2017). *Scientific research methods: Real world research.* Ankara: Ani Publishing.

Rodríguez-Abitia, G., Ramírez-Montoya, M. S., López-Caudana, E. O., & Romero-Rodríguez, J. M. (2021). Factores para el desarrollo del pensamiento computacional en estudiantes de pregrado. *Campus Virtuales*, *10*(2), 153-164. Available at: http://uajournals.com/ojs/index.php/campusvirtuales/article/view/893/490 Access date: 1.12.2021.

Román-González, M., Pérez-González, J. C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior*, *72*, 678-691. https://doi.org/10.1016/j.chb.2016.08.047

Rosenshine, B., & Meister, C. (1992). The use of scaffolds for teaching higher-level cognitive strategies. *Educational leadership*, *49*(7), 26-33. Available at: http://www.formapex.com/telechargementpublic/rosenshine1992a.pdf Access date: 15.12.2021.

Sarı, U., Pektaş, H. M., Şen, Ö. F., & Çelik, H. (2022). Algorithmic thinking development through physical computing activities with Arduino in STEM education. *Education and Information Technologies*, 1-21. https://doi.org/10.1007/s10639-022-10893-0

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142-158. https://doi.org/10.1016/j.edurev.2017.09.003

Strauss, A., & Corbin, J. M. (1997). *Grounded theory in practice*. Sage.

Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. *Computers & Education*, *148*, 103798. https://doi.org/10.1016/j.compedu.2019.103798

Tsalapatas, H., Heidmann, O., Alimisi, R., & Houstis, E. (2012). Game-based programming towards developing algorithmic thinking skills in primary education. *Scientific Bulletin of the Petru Maior University of Targu Mures*, *9*(1), 56-63.

Usta, E. & Korkmaz. Ö., (2010). Difficulties experienced by students while creating algorithms according to different teaching practices. *4th International Computer and Instructional Technology Symposium (ICITS 2010),* Konya 24-26 September. ISBN: 978-605-61434-2-7 pp. 1029-1034.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127-147. https://doi.org/10.1007/s10956-015-9581-5

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

Wing, J. (2011). Research notebook: Computational thinking—What and why. *The Link Magazine, 6,* 20-23. Available at: https://people.cs.vt.edu/~kafura/CS6604/Papers/CT-What-And-Why.pdf Access date: 13.12.2021.

Wohl, B., Porter, B. & Clinch, S. (2015). Teaching Computer Science to 5-7-year-olds: An initial study with Scratch, Cubelets and unplugged computing. *WiPSCE '15: Proceedings of the Workshop in Primary and Secondary Computing Education,* London, United Kingdom. https://doi.org/10.1145/2818314.2818340

Yadav, A., Gretter, S., Good, J., & McLean, T. (2017). Computational thinking in teacher education. In: Rich P., Hodges C. (eds) *Emerging Research, Practice, and Policy on Computational Thinking. Educational Communications and Technology: Issues and Innovations.* Springer, Cham. https://doi.org/10.1007/978-3-319-52691-1_13

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. *ACM Transactions on Computing Education (TOCE)*, *14*(1), 1-16. https://doi.org/10.1145/2576872

Yağcı, M. (2018). A study on computational thinking and high school students' computational thinking skill levels. *International Online Journal of Educational Sciences*, *10*(2), 81-96. https://doi.org/10.15345/iojes.2018.02.006

Yıldırım, A., & Şimşek, H. (2011). *Qualitative research methods in the social sciences.* Ankara: Seçkin Publishing.

Yin, R.K. (2014). *Case Study Research Design and Methods (5th ed.)*. Thousand Oaks, CA: Sage.