



# International Journal of Educational Methodology

Volume 6, Issue 1, 161 - 183.

ISSN: 2469-9632

<http://www.ijem.com/>

## Developing an Instructional Design for the Field of ICT and Software for Gifted and Talented Students\*

Yunus Emre Avcu\*

Balikesir Sehit Prof. Dr. Ilhan Varank Art and Science Center,  
TURKEY

Kemal Oguz Er

Balikesir University, TURKEY

Received: November 8, 2019 ▪ Revised: January 2, 2019 ▪ Accepted: February 11, 2020

**Abstract:** This study aimed to develop an instructional design that focuses on programming teaching for gifted and talented students and to investigate its effects on the teaching process. During the development of the instructional design; the steps of Morrison, Ross and Kemp Instructional Design Model were followed. Embedded experimental design, one of the mixed-method research designs, was used in the modeling of the study. The participants consisted of students studying at the Science and Art Center (BILSEM) (experimental group: 13 girls and 12 boys, control group: 10 girls and 15 boys). While the instructional design developed by the researchers was applied to the gifted and talented students in the experimental group, the standard activities used in Information Technologies and Software Courses at BILSEM were applied to the gifted and talented students in the control group. "Computational Thinking Scale (CTS)", "Torrance Creative Thinking Test (TCTT-Figural)" and "Computer Programming Self-Efficacy Scale (CPSES)" were used to collect the data of the quantitative phase of the study. Qualitative data were gathered by using interview form, observation forms, and design thinking rubric. Two-Factor ANOVA Test, Bonferroni Adjustment Multiple Comparisons Test, and interaction graphs were used to analyze quantitative data while qualitative data were analyzed by content analysis. The quantitative results of the research showed that the instructional design was effective on students' computational thinking and creative thinking skills, but not on programming self-efficacy. Qualitative findings revealed that the instructional design helped the students learn the computational concepts, use computational applications, and develop computational-perspectives. Also, students improved their design thinking skills to a certain level and expressed that they enjoyed the design thinking process, learned the course content, and experienced some difficulties.

**Keywords:** *Instructional design, computer programming, gifted and talented students, Science and Art Center (BILSEM).*

**To cite this article:** Avcu, Y. E., & Er, K. O. (2020). Developing an instructional design for the field of ICT and software for gifted and talented students. *International Journal of Educational Methodology*, 6(1), 161-183. <https://doi.org/10.12973/ijem.6.1.161>

### Introduction

The most general objective of the information age is to educate individuals who can use technology individually or as a group and think correctly to solve the problems they face (Gulbahar, 2018). Beyond professional expectations, computer science education appears as an educational process in which individuals' thinking styles and production skills are shaped to achieve this goal (Kert, 2018a). International Society for Technology in Education [ISTE] (ISTE, 2016) has included computational thinking, creative thinking, design thinking, and creative problem-solving skills through cooperative participation as standards for learning the competencies targeted in computer science education. All these skills are the structures that form the basis for programming skills that can be expressed as the process of producing a problem in a way that technological devices can understand it (Karaman & Kursun, 2018). Programming is a powerful cognitive tool in terms of meeting all criteria of cognitive tools (Jonassen, 2000).

Cognitive, affective, and social gains can be achieved when programming through instructional designs was used as a cognitive tool. Programming teaching has positive effects on cognitive variables such as problem-solving, computational thinking, creative thinking, algorithmic and critical thinking, logical inquiry (Akcaoglu & Koehler, 2015; Calao et al., 2015; Dogan & Kert, 2016; Lye & Koh, 2014, Kim & Kim, 2016). Also, learners can acquire some affective

\* This study was conducted by the first author from his doctoral thesis under the supervision of the second author and funded by Balikesir University (2018/066).

**\*\* Corresponding author:**

Yunus Emre Avcu, Balikesir Sehit Prof. Dr. Ilhan Varank Art and Science Center, Balikesir, Turkey. ✉ [yunus1099@hotmail.com](mailto:yunus1099@hotmail.com)



variables such as programming self-efficacy, interest-desire, motivation, self-confidence and attitudes as well as social competencies through programming teaching (Begosso & da Silva, 2013; Durak, 2016; Gunbatar & Karalar, 2018; Kalelioglu & Gulbahar, 2014). Computational thinking among the cognitive skills which can be obtained through programming teaching is spectacular thanks to both its scope of thinking and its emphasis on the necessity for everyone to have (Lockwood & Mooney, 2017; Lye & Koh, 2014; Wing, 2011). During the computational thinking process, learners use their skills of creative thinking, algorithmic thinking, critical thinking, and problem-solving through creating artifacts (Grover & Pea, 2013; ISTE, 2016). Among the affective behaviors gained through programming teaching, programming self-efficacy has a special significance. It is because individuals who have a higher degree of programming self-efficacy are more eager to solve computational problems and more successful at programming (Kong, 2019). From the perspective of social achievements, programming teaching also addressed as in the framework of socio-cultural constructivism and creative programming activities. During the process of creative programming, learners provide solutions to the unstructured problems with collaborative work together by using their design thinking skills (Romero et al., 2017).

Many countries such as Austria, Czech Republic, Estonia, Great Britain, Finland, Israel, and Turkey has realised the importance of programming teaching in cognitive, affective, and social aspects and the curricula of these countries have been updated in a way that they include programming skills (Balanskat & Elgelhardth, 2015; Kalelioglu, 2018). Thus, it is aimed to foster the development of the 21st-century skills, to meet the need for computer science-based workforce in different sectors (Balanskat et al., 2017) as well as to discover and educate gifted and talented students in the field of computer science. Undoubtedly, to identify gifted and talented students in the field of computer science and to provide appropriate educational services to them are the most effective methods in the long term during the transition process from being a society that consumes technology to a society that produces it (Ongoç & Sozel, 2018).

Science and Art Center (BILSEM), which is a state institution, has the dominant role in educating the gifted and talented students in Turkey. Free educational services are provided to students who are enrolled in BILSEMs after some individual assessments. In BILSEMs, students are taken into Adaptation, Support, Individual Ability Recognition (IAR), Development of Special Abilities (DSA), and Project Production (PROJECT) programs respectively. In the IAR Program, the areas where the students are talented are discovered, and the students are guided to these fields. Students take courses in many fields such as Information Technologies and Software, Science, Mathematics, and Social Studies in the IAR Program (Ministry of National Education [MoNE], 2016). At BILSEMs, emphasis on education in the fields of art, science, and mathematics (Geckil, 2012) has a negative effect on gifted and talented students in a special and different field such as information technologies and software (Keskin, 2006). However, talents can be developed further if gifted and talented students are discovered and supported in the field of computer science (Siegle, 2004). Also, these students can be encouraged to contribute to the field (Colluoglu Gulen, 2014).

MoNE has been developing framework programs and activity books for the gifted and talented students who are interested in computer science and aiming at developing themselves in this field, and also in this work, programming achievements are specially featured (MoNE, 2017). However, the fact that these framework programs and activity books designed for these programs are technology-oriented, and they have pedagogical deficiencies can be criticized. Even the aim of developing computational thinking, creative thinking, and design thinking skills; and improving affective skills related to programming emphasized in framework programs and activity books, achievements regarding the development of these skills have not been stated in documents. Also, it is because the use of tool-oriented approaches in programming teaching with uncertain acquisition relationships in technology-oriented teaching processes causes teaching to be inefficient (Kert, 2018b). Therefore, it is extremely important for the gifted and talented students to develop instructional designs whose pedagogical deficiencies are made up for, that focus on achievements and provides rich learning experiences. Although strong evidence is needed for the educational practices applied to gifted students, the relevant literature is lacking in terms of extensive empirical studies (Plucker & Callahan, 2014). In the literature review, it is seen that there are instructional designs developed for teaching language (Kaplan-Sayi, 2013) by application of different design models and for teaching Science-Math to the gifted and students (Ayverdi, 2018; Ozcelik & Akgunduz, 2018; Tyler-Wood et al., 2000; VanTassel-Baska et al., 1998). The number of specially developed instructional designs that focus on programming education for gifted and talented students is limited (Durak, 2016; Kim et al., 2013; Shin et al., 2013). In previous studies, only the block-based programming approach preferred in programming teaching.

In the scope of this research, it is aimed to develop an instructional design that is focused on the learner and used tool-focused programming pedagogies for gifted and talented students and to investigate this instructional design's effects on the process of teaching. Programming pedagogies also used in the teaching of computational thinking skills (Kert, 2018b). Creative thinking and design thinking addressed as skills that can be improved with creative programming, and at the same time, processes of programming teaching are enriched by using design thinking as a learner-centered approach. With this study, it is expected to contribute a) to the individual in terms of acquiring thinking skills and acknowledging special talents in computer science, b) to the society in terms of educating gifted and talented students by developing technology to support Turkey's development goals, c) to the educational science and computer science

fields by i. addressing the stages of instructional design development in detail by using a model, ii. providing evidence in terms of applicability of different pedagogies and effectiveness of a design, iii. fulfilling the gap within the literature.

## Research Background

### *Research Problem*

The problem of this research is defined as “What are the effects of the instructional design developed by using Morrison, Ross, and Kemp Model in the field of information technologies and software for gifted and talented students on the teaching process?”

### *Sub-problems*

1. Does the instructional design affect the computational thinking, creative thinking and programming self-efficacy scores of gifted and talented students?
2. What are the views of gifted and talented students about computational thinking before and after the implementation of the instructional design?
3. What are the views of gifted and talented students about design thinking before and after the implementation of instructional design?
4. What is the gifted and talented students' state of using computational thinking and design thinking in terms of teacher observations during the experimental process?
5. How do design thinking worksheets, note sheets used in the idea-making process, and prototypes developed by students reflect the design thinking process of gifted and talented students?

### *Computational Thinking (CoT)*

Computational Thinking (CoT) is “a process of thinking that involves formulating problems and solutions for them to present the solutions in a form that can be effectively implemented by an information processing unit (human, machine, robot)” (Wing, 2011, p.1). Formulation problems require the use of cognitive processes such as modularizing, abstracting and generalization. According to Garcia-Valcarcel Munoz-Repiso and Caballero-Gonzalez (2019), CoT is the capacity and ability to solve problems using the basic principles of programming and computer science. CoT has three basic elements: computational concepts, computational practices, and computational perspectives. Computational concepts are the concepts used in programming. Computational practices deal with the practices that learners apply when they produce something by programming. On the other hand, computational perspectives constitute the affective dimension of CoT and are the insights developed by the learners against themselves, their relations with others and the technological world (Brennan & Resnick, 2012). CoT can be taught by computer-free (unplugged) programming, physical programming, block-based programming, text-based programming, and interdisciplinary applications (Kandemir, 2018; Loockwood & Mooney, 2017; Weinberg, 2013).

### *Creative Thinking (CT)*

Creativity is a context-specific process in which a new, appropriate, and useful solution is developed individually or collaboratively by a reference group (McGuinness & O'Hare, 2012). It is enough for a student to create an idea or product that is new, appropriate, and useful for his/her creativity in the classroom (Starko, 2014). In a learning environment in which creativity is supported, creative thinking skills can be developed in many contexts (Baer & Kaufman, 2012; Orhon, 2014). One of these contexts is computer science education, which focuses on learner-oriented production processes. In computer science education, it is aimed to reveal the characteristics of the learners as a designer, developer, and active participant. Concrete products and software-based content development activities allow learners to develop their creativity (Kert, 2018a; Romero et al., 2017). Learners can involve in the creative thinking process by producing content utilizing digital tools individually or in teams (Romero, Laferriere & Power, 2016).

### *Computer Programming Self Efficacy (CPSE)*

Computer programming self-efficacy (CPSE) reflects the individual's perception and assessment of his/her ability to solve computational problems using programming knowledge and skills. Individuals with high CPSE are more willing to apply their knowledge and use their skills to solve computational problems (Kong, 2017). Low CPSE is an obstacle that affects the performance of individuals in programming teaching (Hongwarittorn & Krairit, 2010). CPSE is a key variable to be successful in programming activities (Yildiz-Durak, Karaoglan-Yilmaz & Yilmaz, 2019). Also, CPSE is considered a reflection of computational perspectives (Roman-Gonzalez et al., 2019). Determining the levels of CPSE helps to comment on the programming success of the individuals (Askar & Davenport, 2009; Ramalingam, LaBelle & Wiedenbeck, 1998).

*Design Thinking (DT)*

Design (DT) is a human-centered approach that provides creative and innovative solutions to problems using design tools and mindsets (Carroll et al., 2010). Using DT steps (empathy, defining, generating ideas, developing prototypes, testing) creative solutions can be produced for personal, social, and commercial problems (Bootcamp Bootleg D.School, 2011; Kelley & Kelley, 2013). DT is also used in areas other than the design field (such as business, engineering, education, etc.) as it encourages creativity and innovation with an empathic, flexible, and iterative approach (Henriksen, Richardson & Metha, 2017; Lor, 2016). It is known that DT is used as a method, process, and skillset in the field of education together with the search of a teaching strategy suitable for 21<sup>st</sup> century skills, work habits, and character traits (Aflatoony & Wakkary, 2018; Carroll, 2014; Dukes & Koch, 2012). In creative programming activities, DT is used as a human-centered teaching method. In the creative programming process, learners use DT as a skill and method. They understand the nature of an open-ended problem, empathize, identify the problem, and configure, develop, and improve the program in a way that is will be a new, convenient, and useful solution to the problem (Romero et al.,2017).

**Methodology**

*Research Design*

Embedded experimental design, one of the mixed-method research designs, was used in the modeling of the study. The embedded experimental design emerges when the researcher embeds qualitative data into experimental designs (Creswell & Plano Clark, 2014). The research problem in this study was studied by adding qualitative data to the experimental design before, during, and after the experiment. The research process is given in Figure 1.

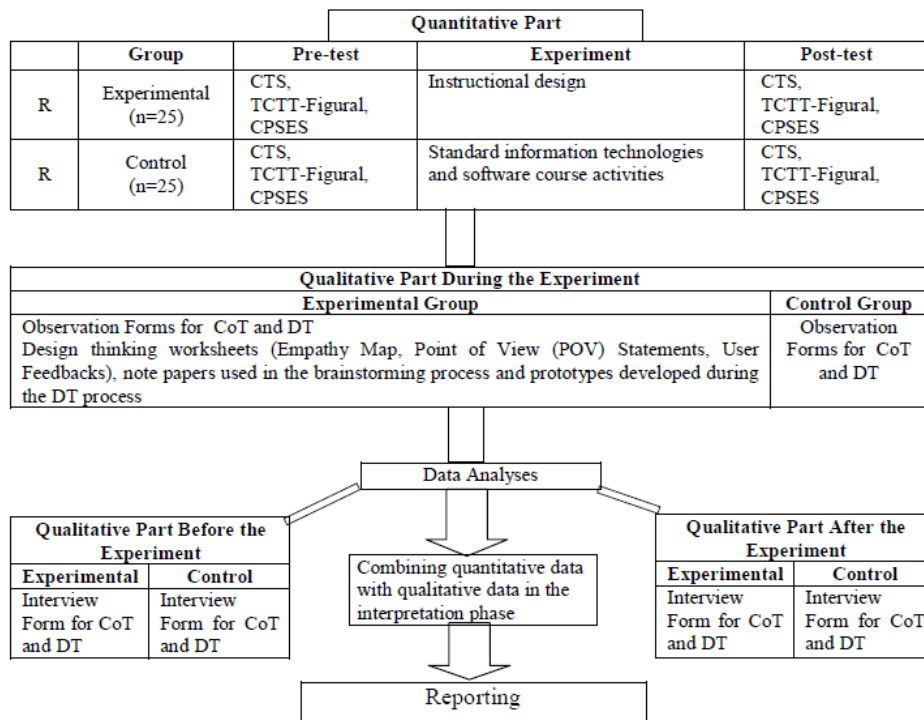


Figure 1. Research Process

In the quantitative part of the research, “Random Design with Pre-test, Post-test, and Control Groups” was used to investigate the effects of instructional design on computational thinking, creative thinking, and computer programming self-efficacy of gifted and talented students. In this design, two groups, experimental and control were formed by random assignment from the determined experimental subject pool. Then, the measurements of the variables in the experimental and control groups regarding the dependent variables were taken. The experimental process whose effect to be measured was given to the experimental group but not to the control group. The measurements of the dependent variables of the experimental subjects in the experimental and control groups were obtained by applying the same or equivalent scale forms (Buyukozturk, 2014). In the qualitative part of the research, the views of gifted and talented students about computational and design thinking skills were examined through interviews before and after the experimental process. Their use of computational thinking and design thinking skills was determined thanks to teacher observations. Also, the students in the experimental group were interpreted by examining the design thinking worksheets, the note papers they used in the process of generating ideas and the prototypes they developed.

### Participants

Participants were the gifted and talented students enrolled in IAR program at Balikesir Sehit Prof. Dr. Ilhan Varank Science and Art Center (experimental group: 13 girls, 12 boys, and control group: 10 girls, 15 boys). 8 of the students in the experimental group were at 5th grade, 7 of them were at 6th grade, 3 of them were at 7th grade, and 7 of them were at 8th grade. 5 of the students in the control group were in 5th grade, 10 of them were in 6th grade, 3 of them were in 7th grade, and 7 of them were in 8th grade. 25 students were randomly assigned to experimental and control groups. In BILSEM, students are grouped by program level, not by grade level. The ages of students in the IAR program can be different as + -2 years from each other. Students at the same program level at BILSEM are provided education according to their interests and needs.

### Data Collection Tools

Quantitative data was gathered by using the Computational Thinking Scale (CTS), Torrance Creative Thinking Test (TCTT-Figural), and Computer Programming Self-Efficacy Scale (CPSES). Qualitative data was gathered with interview form, observation forms, and DT Rubric developed by the researchers.

#### Quantitative Data Collection Tools

*Computational Thinking Scale (CTS):* CTS was developed by Korkmaz et al. (2015) to measure the computational thinking skills of secondary school students. In total, there were 22 items and five factors consisting of creativity (4 items), algorithmic thinking (4 items), collaboration (4 items), critical thinking (4 items), and problem-solving (6 items). The items of the 5-point Likert scale are graded as never (1), rarely (2), sometimes (3), usually (4), always (5). The Cronbach's Alpha reliability coefficient for the scale was .809. Cronbach's Alpha values of the factors ranged from .640 to .867. Factor-total correlation of all factors in the scale ranged between .655 and .842, and t values were significant ( $p < .001$ ). These results showed that the level of each item's serving its purpose was high.

*Torrance Creative Thinking Test (TCTT-Figural) (A and B Forms):* TCTT-Figural is often used to identify creative individuals around the world. This test was first developed by Torrance (1966), and then norm studies were carried out five times. The norms of the latest version of TCTT 2007 were based on data from 70093 people. Two different score types based on norm and criteria were used in the calculation of the scores obtained from TCTT-Figural A and B forms. The scores of the sub-dimensions such as fluency, originality, the abstraction of titles, enrichment, and resistance to early completion were calculated as norm-based. Criteria-based scores were obtained by summing the scores of the sub-dimensions under the Creative Strengths Checklist. The sum of the scores based on norms and the scores based on criteria gives the creativity score.

*Computer Programming Self-Efficacy Scale (CPSES):* CPSES was developed by Kukul et al. (2017) to measure the level of programming self-efficacy of secondary school students. It consists of 31 items and a single factor. The item loads of the items in the scale ranged from .47 to .71, and the variance explanation percentage of the single factor scale was 41.15%. The 5-point Likert-type scale with no reverse item was rated as "Strongly Agree (1)" and "Strongly Agree (5)". At the end of the reliability analysis, the Cronbach's Alpha reliability coefficient was calculated as .95. In addition, the Spearman-Brown reliability coefficient was calculated as  $r = .96$  with the split-half method. These results showed that the internal consistency of the measurement tool was high. In order to determine the construct validity, exploratory and confirmatory factor analyses were performed and according to the results obtained from the analyses, the scale was found to be valid and reliable.

#### Qualitative Data Collection Tools

*Interview Form:* An interview form was formed by the researchers to examine the viewpoints of the students in the study group towards computational and design thinking skills. The form contains 8 open-ended questions for computational thinking, 5 open-ended questions for design thinking, alternative questions and probes for each question. After the interview form was prepared, opinions about the form were obtained from 3 experts, and the form was finalized in line with the suggestions from those experts. Interviews were carried out by the researcher with 4 students each from the experiment and control groups, before and after the experimental process at BILSEM. These interviews lasted 23-40 minutes on average. The questions directed to the students were about describing computing concepts (sequence, loops, events, parallelism, conditionals, operators, variable, list and function), how they are structuring a programming project, how they are finding their mistakes and correcting them while programming, interacting with other people in the programming process, their thoughts about expressing themselves by programming. For DT; questions such as how are they going to structure a design project, what kinds of tools they will use for prototyping, what they think about possible problems may be encountered during design process and what they think about the personality traits that a member of the design team should inherit are directed to students.

*Observation Forms:* The computational thinking observation form was developed by researchers using the online platforms "ScratchEd" and "d school K12 Lab". The number of times that students demonstrate behaviors that meet the criteria are marked in the frequency section of the related items in the computational thinking observation form. In the

DT observation form, the level of the students regarding the DT levels is marked on the cells in the observation form. Experimental observations were made during the pilot implementation of the instructional design and whether the criteria in the observation forms are performed was tested during this pilot implementation and the opinions of 3 experts were also obtained. The observations lasted 40 hours, 20 hours each in both experimental and control groups. The validity of the observation forms was ensured by reviewing the literature and obtaining expert opinion. "Cohen's Kappa Coefficient" was calculated for the reliability of the computational thinking observation form and found to be .66 for the experimental group and .76 for the control group. The agreement between the observers for the DT observation form was determined by calculating the "Weighted Kappa Coefficient" and found to be .61 for the experimental group and .67 for the control group. It can be said that there is a good level of harmony between observers for both observation forms and both groups (Landis & Koch, 1977, p.165).

*DT Rubric:* Design thinking worksheets (Empathy Map, POV statements, User Feedbacks) (Annex-2), note papers used in the brainstorming process, and prototypes developed during the DT process were examined with DT Rubric (Annex-2) developed by the researchers. Performance tasks are defined as DT tasks. In addition, the appropriateness of the performance tasks to the age level of secondary school students and in terms of Turkish language rules were examined by taking teachers' opinions. DT tasks in instructional design were as follows: 1. DT Task: to redesign the experience of playing computer games, 2. DT Task: to redesign the healthy eating experience, 3. DT Task: to redesign the water usage experience, 4. DT Task: to redesign the experience of living safely, 5. DT Task: To redesign the learning experience gained in science and technology. Five criteria have been identified that focus on the steps of the DT process to be used in determining performance. The performance levels were determined from the weakest to the most competent by attaining 1 point for the student with the weakest and 4 points for the highest performance. After the opinions of 8 students, 2 experts and 1 teacher who participated in the pilot design of the instructional design were received, the DT rubric was given its final form. The intra-class correlation coefficient (ICC) was calculated for the agreement between the raters and was found to be .825. It can be said that there is a "good level of fit" among the raters of the design thinking rubric (Koo & Li, 2016; p.158).

#### *Forming an Instructional Design and Implementation Process*

The instructional design was based on the instructional design model by Morrison et al. (2012), and its phases were followed. In Figure 2, details are given regarding stages of instructional design.



Figure 2. Stages of Instructional Design.

The actual implementation was realized as a project in the summer term of the 2017-2018 academic year. The application took a total of 74 hours. An average of three or four hour-implementation was carried out per day, and this duration was decided in order not to divide the activities. The pre-tests were applied to the students before the instructional design started. During the last day of the training, the final tests were performed, and a photo show and certificate ceremony was held for the participants during the implementation process. While the instructional design developed was applied in the experimental group, the control group studied the standard information technologies and software activities. The standard activities applied in the control group were planned for teaching the same units in the experimental group. The researcher, instructor, and observer in the experimental group took part in the

implementation of the activities in the control group. The application in the control group was carried out in the form of a summer-term project as in the experimental group. Quantitative and qualitative approaches were used together in order to measure the impact of instructional design and product evaluation was realized in this way. In the context of confirmatory evaluation, the long-term impact of the design wasn't investigated.

#### Data Analysis

Quantitative and qualitative data were analyzed using SPSS and NVivo programs. In the analysis of quantitative data, Two-Factor ANOVA Test for Repeated Measurements, Bonferonni Compatible Multiple Comparisons Test, and interaction graphs were used. The normal distribution of the data set was examined by normality tests, descriptive values, and Q-Q graphs. It was understood that the data showed normal distribution. ANOVA test assumptions were examined, and the applicability of the test was approved. The analyses were conducted with 95% confidence interval with the significance level  $\alpha = .05$ . The data obtained from the interviews were analyzed by content analysis. Data collected through interview forms were examined under themes and sub-themes. The data is encoded by two encoders to ensure reliability between encoders. In coding for reliability by two researchers, the consistency between coders was calculated as 89% (reliability = (consensus number) / (total consensus + consensus number)) (Miles & Huberman, 1994). Direct quotations were taken from the interviews with the students and the quotations were coded as S1-S4, experimental-control group (E and C) and female-male (F and M).

### Findings

#### Findings of the First Sub-Problem

In order to test the effect of instructional design on the CoT, CT, and CPSE scores of the gifted and talented students, a two-factor ANOVA test was performed. Pre-test and post-test mean scores and standard deviation values of CoT are presented in Table 1.

Table 1. Findings on the Pre and Posttest Mean Scores of CoT

Groups	Pre and Post-test Mean Scores	N	$\bar{X}$	SD
Experimental Group	Pretest	25	88.76	9.41
	Posttest	25	94.52	6.04
Control Group	Pretest	25	85.57	8.47
	Posttest	25	88.68	8.43

Table 1 shows that there is an increase in the scores of both groups about CoT; however, the increase in points in the experimental group was more evident. ANOVA test was applied to test whether the increase in scores was statistically significant, and the results are given in Table 2.

Table 2. Findings on Repeated Measures ANOVA of CoT

Source of Variance	Sum of Squares	df	Mean Square	F	p	$\eta^2$
<b>Between Groups</b>	6446.536	49				
Group (Experimental / Control)	509.377	1	509.377	4.118	.041	.080
Error	5937.159	48	123.691			
<b>Within Groups</b>	1031.22	50				
Test (pretest and posttest)	490.517	1	490.517	47.397	.001	.497
<b>Group*Test</b>	<b>43.947</b>	<b>1</b>	<b>43.947</b>	<b>4.246</b>	<b>.040</b>	<b>.081</b>
Error	496.756	48	10.349			
Total	7477.756	99				

The joint effect of the group and test are given in Table 2 shows that there was a significant difference between the groups depending on the measurements ( $F(1,48) = 4.246$ ;  $p < .05$ ,  $\eta^2 = .081$ ). When the Eta squared value is examined, it is understood that the joint effect of the group and the test has a medium effect size. It was concluded that instructional design had a significant effect on CoT scores. In order to compare the differences between the mean scores of the experimental and control groups, "Bonferonni Adjustment Multiple Comparisons Test" was applied, and the results are presented in Table 3.



Table 3. Bonferroni Adjustment Multiple Comparisons Test Results of CoT

		Bonferonni Test Results			
		Experimental		Control	
		Pre-test	Post-test	Pre-test	Post-test
		Mean Difference (I-J)	Mean Difference (I-J)	Mean Difference (I-J)	Mean Difference (I-J)
Experimental	Pre-test		-5.755*	3.188	
	Post-test	5.755*			5.840*
Control	Pre-test	-3.188			-3.104*
	Post-test		-5.840*	3.104*	

\*p&lt;.05

When Table 3 is examined, it is seen that the difference between pre-test mean and post-test mean of experimental and control groups is significant. In terms of pre-test scores, the mean scores of the experimental and control groups do not differ significantly. The mean post-test scores of the experimental group and control group showed a significant difference in favor of the experimental group ( $p < .05$ ). The interaction graph is presented in Figure 1.

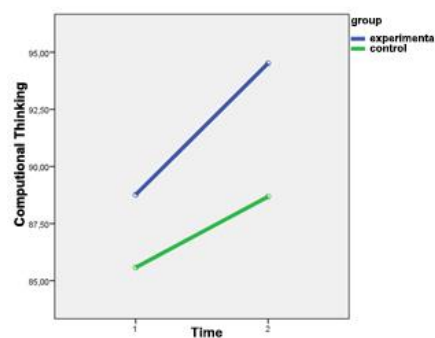


Figure 3. Interaction Graph of CoT

When Figure 3 is examined, it is seen that there is an increase in the mean scores of both groups; however, the increase in the experimental group is more pronounced than the control group. In order to test the effect of instructional design on CT scores, TCTT-Figural A and B equivalent forms were applied to experimental and control groups. Pre-test and post-test mean scores and standard deviation values related to CT are given in Table 4.

Table 4. Findings on the Pre and Posttest Mean Scores of CT

Groups	TCTT-Figural Pre and Post-test Scores	N	$\bar{X}$	SD
Experimental Group	Pretest	25	64.36	13.94
	Posttest	25	88.56	18.35
Control Group	Pretest	25	66.56	10.98
	Posttest	25	69.88	11.43

When Table 4 is examined, it is seen that the increase in score in the experimental group is more pronounced than the control group. ANOVA test was performed in order to examine the increase in the scores of the experimental and control groups statistically, and the results are given in Table 5.

Table 5. Findings on Repeated Measures ANOVA of CT

Source of Variance	Sum of Squares	df	Mean Square	F	p	$\eta^2$
<b>Between Groups</b>	17149.44	49				
Group (Experimental / Control)	1697.440	1	169.440	5.273	.026	.099
Error	15452.000	48	321.917			
<b>Within Groups</b>	10.7930	50				
Test (pretest and posttest)	4733.440	1	4733.440	68.133	.001	.587
<b>Group*Test</b>	<b>2724.840</b>	<b>1</b>	<b>2724.840</b>	<b>39.221</b>	<b>.001</b>	<b>.450</b>
Error	3334.720	48	69.473			
Total	27942.44	99				

When Table 5 is examined, it is concluded that the group and test common effect is significant ( $F_{(1,48)}=39.221$ ;  $p < .05$ ,  $\eta^2=.450$ ). When the eta squared value is examined, it is observed that the joint effect time of experimental intervention

has a large effect size. In light of the findings, it can be said that instructional design has an effect on creative thinking scores. “Bonferonni Adjustment Multiple Comparisons Test” results are given in Table 6.

Table 6. Bonferonni Adjustment Multiple Comparisons Test Results of CT

		Bonferonni Test Results			
		Experimental		Control	
		Pre-test	Post-test	Pre-test	Post-test
		Mean Difference	Mean Difference	Mean Difference	Mean Difference
		(I-J)	(I-J)	(I-J)	(I-J)
Experimental	Pre-test		-24.200*	2.200	
	Post-test	24.200*			-18.680*
Control	Pre-test	-2.200			-3.320*
	Post-test		18.680*	3.320*	

\*p<.05

According to Table 6, the difference between the pre-test mean and post-test mean scores of the experimental and control groups is significant, while the mean scores of the experimental and control groups do not differ significantly in terms of the pre-test scores. The mean post-test scores of the experimental and control groups show a significant difference (p <.05). The graph of interaction is presented in Figure 2.

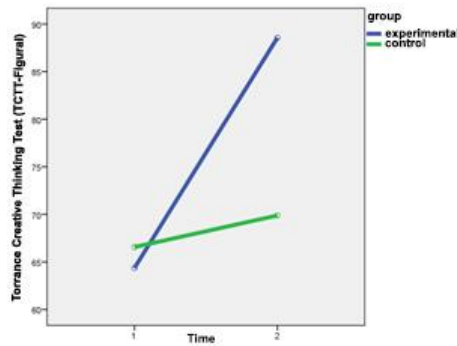


Figure 4. Interaction Graph of CT

Figure 4 shows that there are large differences between the CT post-test scores of the experimental and control groups and the post-test score of the experimental group is higher than the control group. When the statistical findings and the interaction graph are analyzed together, it can be said that instructional design is effective in improving the skills of gifted and talented students.

Table 7. Findings on the Pre and Posttest Mean Scores of CPSE

Groups	PT Pre and Post-test Scores	N	$\bar{X}$	SD
Experimental Group	Pretest	25	134.23	20.97
	Posttest	25	142.96	8.67
Control Group	Pretest	25	132.07	17.58
	Posttest	25	134.99	12.38

When Table 7 is examined, it is understood that there are differences between the pre-test and post-test measurements of the experimental group and the post-test scores of the experimental and control groups. The ANOVA test was applied to test whether the increase in score, is statistically significant, which can also be understood from descriptive statistics, and the results are given in Table 8.

Table 8. Findings on Repeated Measures ANOVA of CPSE

Source of Variance	Sum of Squares	df	Mean Square	F	p	$\eta^2$
<b>Between Groups</b>	20311.22	49	1051.016			
Group (Experimental / Control)	640.799	1	640.799	1.562	.217	.032
Error	19690.423	48	410.217			
<b>Within Groups</b>	4839.924	50	1137.293			
Test (pretest and posttest)	847.858	1	847.858	10.762	.002	.183
<b>Group*Test</b>	<b>210.656</b>	<b>1</b>	<b>210.656</b>	<b>2.674</b>	<b>.109</b>	<b>.053</b>
Error	3781.410	48	78.779			
Total	25151.144	99	2188.309			

The insignificant effect of the group and test common effects given in Table 8 indicates that there is no significant difference between the groups in terms of pre-test and post-test measurements ( $F_{(1,48)}=2.674$ ,  $p>.05$ ,  $\eta^2=.053$ ). The interaction graph is given in Figure 3.

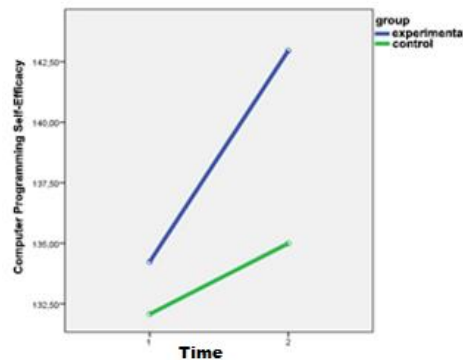


Figure 5. Interaction Graph of CPSE

In the interaction graph given in Figure 5, although there is a significant difference between the mean post-test scores of the experimental and control groups, it is not possible to say that the increase in the mean scores result from the experimental procedure as the group test common effect is not significant in the ANOVA test. As a result, it can be said that instructional design has no significant effect on the programming self-efficacy of gifted and talented students.

#### Findings of the Second Sub-Problem

Semi-structured interviews were conducted with four students in the experimental and control groups before and after the experimental procedure, and the results of the content analysis are presented in Table 9.

Table 9. Frequency Distribution of the interviews with the Students in Experimental Group about CT

Pre-interviews with Experimental Group		Post-interviews with Experimental Group	
Categories	Codes	Categories	Codes
Computational Concepts	Lack of knowledge (f:15), loop (f:4), use of conditionals (f:2), operators (f:1), parallelism (1), sequencing (f: 3), Data (variable (f:1))	Computational Concepts	Loop (f:4), use of conditionals (f:4), event control (f:4), operators (f:4), parallelism (f:2), sequencing (f:4), Data (variable (f:4), list (f:4) and function (f:4))
Computational Practices	Modularizing (f:3), experimenting and iterating (f:4), debugging (f:7), abstracting (f:1), testing (f:4), data collection (f:3), Reusing and Remixing (using a part (f:2), getting inspiration (f:2), citing (f:4) and using all the parts (f:1))	Computational Practices	Modularizing (f:6), experimenting and iterating (f:5), generalization (f:1), debugging (f:6), automation (f:3), abstracting (f:4), testing (f:3), data collection and analyzing (f:4), Reusing and Remixing (using a part (f:3), getting inspiration (f:1), and citing (f:3))
Computational Perspectives	Connecting (coding for the others (f:3), coding with others (f:4)), expressing (f:2) and questioning (f:2)	Computational Perspectives	Connecting (coding for the others (f:2) and coding with others (f:3)), expressing (f:3) and questioning (f:3)

Students in the experimental group expressed their opinions about computational concepts, practices, and perspectives. When Table 9 is examined, it is seen that the students in the experimental group talk about event control, list, and function after the experimental procedure, unlike their views before the experimental procedure. Also, in the last interviews, students talked about computational applications, automation, and generalization, and mentioned abstraction more. In the last interviews about the computational concepts, it was seen that the students didn't have any lack of information and that the computational concepts and applications are mentioned more. S1-E-F stated in the preliminary interviews that there was a lack of information regarding the concepts of computational thinking: "I have not heard the list. Let's assume that we have a product. We can define the function as the features of our product." In general, S4-E-M denoted in the last interviews: "I learned how the servo motors work in the project, how it turned in different degrees. Let's say there is a walking robot, I can understand how it works from the running principle of servo motors. I will code its walking with the same principle."

Table 10. Frequency Distribution of the interviews with the Students in Control Group about CT

Pre-Interviews with Control Group		Post-Interviews with Control Group	
Categories	Codes	Categories	Codes
Computational Concepts	Lack of knowledge (f:9), loop (f:4), use of conditionals (f:4), operators (f:4), sequencing (f:3), Data ((f:1) and list (f:1))	Computational Concepts	Lack of knowledge (f:7), loop (f:4), use of conditionals (f:2), operators (f:3), parallel processing (f:1), sequencing (f:4), Data (variable (f:4), list (f:3), function (f:1))
Computational Practices	modularizing (f:3), experimenting and iterating (f:2), debugging (f:7), abstracting (f:3), testing (f:3), data collection (f:1), Reusing and Remixing (using some part of it (f:2), getting inspiration (f:1) and citing (f:4)	Computational Practices	Modularizing (f:3), experimenting and iterating (f:2), debugging (f:7), abstracting (f:4), testing (f:2), data collection (f:2), Reusing and Remixing ((f:3), getting inspiration (f:2), citing (f:4))
Computational Perspectives	Connecting (coding for the others (f:3), coding with others (f:4)), expressing (f:3) and questioning (f:1)	Computational Perspectives	Connecting (coding for the others (f:3) and coding with others (f:4)), expressing (f:4)

Students in the control group expressed their opinions about computational concepts, practices, and perspectives. When Table 10 is examined, it is seen that the students' lack of knowledge regarding information processing concepts persists in the last interviews; however, it is seen that parallelism and function concepts are not included in the pre-experiment interviews are included in the post-experiment interviews. In addition, students talked about abstraction as well as reusing and remixing applications after experimental procedures. In the preliminary interviews, S2-C-M defined "modularizing" as: "I first determine my project, look at the elements, what do they do in this project, and I will continue to write their tasks one by one accordingly." S3-C-F mentioned about "abstracting" in the last interview as: "I will organize it by throwing away the unnecessary codes, dead codes. To read the code easily and find the error easily, we select and find the character with error and correct it."

*Findings of the Third Sub-Problem*

The results of the DT obtained from semi-structured interviews with four students before and after the experimental procedure are presented in Table 11 and Table 12.

Table 11. Frequency Distribution of the interviews with the Students in Experimental Group about DT

Pre-Interviews with Experimental Group		Post-Interviews with Experimental Group	
Categories	Codes	Categories	Codes
Design Thinking Process	Researching and exploring (f:2), empathy (f:4), creating ideas (f:2), prototyping (f:2), Problems during the process (lack of material (f:2), not doing the responsibilities (f:1), defining the problem correctly (f:1), team friends' being inconsiderate (f:2), disapproval of design (f:1), technical errors (f:1))	Design Thinking Process	Empathy (f:6), creating ideas (f:4), learning content (f:4), sharing (f:4), prototyping (f:4), Problems during the process (problems in creating ideas (f:1), not doing he responsibility (f:3), disagreement among the team members (f:4), technical errors (f:2) and time limitation (f:1)), enjoying the process (f:2), defining/POV (f:4), testing (f:4), repeating (f:1)
Prototyping Tools	3D printer (f:1), coding (f:1), model cardboard (f:1), robotic materials (f:2)	Prototyping Tools	3D design software (f:4), Daily tools and materials (f:9), coding tools (f:9), robotic materials (f:5), strawbee (f:1)
Characteristics of The Designer	researcher (f:1), hardworking (f:1), reliable (f: 1), leader (f:1), able to define the problem (f:1), patient (f:4), responsible (f:2), with high technical skills (f:4), creative (f:3), able to use the time effectively (f:1)	Characteristics of The Designer	Understanding (f:2), respectful towards other peoples' opinions (f:2), hardworking (f:2), solution-oriented (f:1), values thinking (f:3), empathic (f:1), active listener (f:1), loyal to his/her duty (f:2), can coordinate easily in every environment (f:1), tolerant (f:1), leader (f:1), optimistic (f:1), patient (f:2), even-tempered (f:1), responsible (f:1), team worker (f:2), with high technical skills (f:2) creative (f:2)

Students in the experimental group expressed their opinions about the DT process, prototyping tools and characteristics of the designer. When Table 11 is examined, it is seen that the interviews after the experimental procedure focus on learning content, sharing, enjoying the process, defining/POV, and repetition, unlike the interviews before the experimental procedure. It can be said that the activities carried out for DT in the experimental group affected students' learning the DT process as well as the content. In recent interviews, the use of daily materials as prototyping tools was mentioned. It was only stated in recent interviews that a good designer should be a person who values thinking, empathizes, works within a team, and respects other ideas. S1-E-F expressed the empathy level of the DT process by saying *"I learn their problems by asking others, I get to know them closely and design for them."* The problem of failure to fulfill the responsibility during the DT process was stated by S3-E-M in the last interviews as follows: *"Whether it was the prototyping phase or the software phase, there were people left behind in our team, who dealt with things other than the task and did not want to get into the event."*

Table 12. Frequency Distribution of the interviews with the Students in Control Group about DT

Pre-Interviews with Control Group		Post-Interviews with Control Group	
Categories	Codes	Categories	Codes
Design Thinking Process	Researching and exploring (f:3), empathy (f:3), creating ideas (f:1), planning (f:1), prototyping (f:1), Problems during the process (disagreement (f:1), problems while creating ideas (f:2), arguments among the team members (f:1), disapproval of the design (f:1), technical errors (f:2))	Design Thinking Process	creating ideas (f:2), planning (f:2), prototyping (f:2), Problems during the process (not respecting the opinion of other people) (f:1), not doing the responsibilities (f:2), disagreement among the team members (f:2), technical errors (f:3), working with people at different ages (f:1))
Prototyping Tools	3D design software (f:2), recycling materials (f:1), robotics materials (f:2), construction materials (f:2)	Prototyping Tools	3D design software (f:2), recycling materials (f:1), coding (f:2), construction materials (f:3)
Characteristics of The Designer	Determined (f:1), solution-oriented (f:1), tidy (f:1), educated (f:1), tolerant (f:1), disciplined (f:1), sharing (f:1), planned (f:1), patient (f:1), calm (f:1), team worker (f:2), creative (f:3), helpful (f:1)	Characteristics of The Designer	Knowledgeable (f:1), active listener (f:1), sociable (f:1), objective (f:1), respectful (f:1), even-tempered (f:1), responsible (f:1), easy going (f:1), creative (f:1), helpful (f:1), serious if needed (f:1)

Students in the control group expressed their opinions about the DT process, prototyping tools and characteristics of the designer. When Table 12 is examined, it is seen that, unlike the preliminary interviews, the students stated in the last interviews that working with people who are not at the same age during the DT processes may be a problem. It was stated by students that 3D design software, coding, construction, and recycling materials could be used as prototyping tools. Regarding design features, the vast majority of students are observed to emphasize on the designer's being creative, harmonious, respectful, and effective listener. In the preliminary interview, S1-C-F said: *"The design process starts with looking at previous designs. If I am a designer, I go to that class once, note down the missing parts and what to change."* In the last interview, S3-C-F denoted that the designer should be creative and harmonious by saying: *"He/she should be creative and harmonious. I can fill in the missing parts, but my teammates just need to be creative and harmonious."*

#### Findings of the Fourth Sub-Problem

According to the observation results, the behaviors that enable the students in the experimental and control groups to demonstrate their CoT and DT skills are presented in Table 13 and Table 14.

Table 13. Behaviors that Enable Students to Demonstrate Their CoT Skills

Behaviors		Groups	
		Experimental n=25 (f)	Control n=25 (f)
1. Experimenting and Iterating	The student.....		
	Builds a project step by step	135	121
	Tries the step as soon as he/she builds it	134	118
	Revises his/her steps in line with the trial results.	133	115
	Tries new ways or things to realize the steps	128	88
	<b>Experimenting and Iterating Total (f)</b>	<b>530</b>	<b>442</b>
2. Testing and Debugging	Observes what happens when he/she runs the code	137	105
	Describes what is different from what she has expected when his/her code doesn't work	129	115
	Reads through the codes to investigate the cause of the problem	133	95
	Makes changes in the codes and test to see what happens	125	98
	Considers other ways to solve the problem	107	39
		<b>Testing and Debugging Total (f)</b>	<b>631</b>
3. Reusing and Remixing	Finds ideas and inspiration from other projects and creates new ideas	47	22
	Selects a piece of another project, and adapts it for his/her project	17	15
	Modifies an existing project to improve or enhance it	18	12
	Gives reference to people whose work he/she gets used or is inspired by	13	4
		<b>Reusing and Remixing Total (f)</b>	<b>95</b>
4. Abstracting and Modularizing	Decides which characters/components to use and their roles	100	96
	Decides which codes to write and their roles	99	74
	Organizes the scripts in ways that make sense to himself/herself and others	86	49
		<b>Abstracting and Modularizing Total (f)</b>	<b>285</b>
<b>Total (f)</b>		<b>1541</b>	<b>1116</b>

When Table 13 is examined, it is seen that the students in the experimental group have exhibited a total of 1541 behaviors and while the ones in the control group have shown 1116 behaviors in terms of CoT. Accordingly, it can be said that the students in the experimental group exhibit more CoT behaviors than the ones in the control group.

Table 14. Behaviors that Enable Students in the Experimental and Control Groups to Demonstrate Their DT Skills

Student Behaviors	Groups									
	Experimental n=25 (f)					Control n=25 (f)				
	Level 1	Level 2	Level 3	Level 4	Total	Level 1	Level 2	Level 3	Level 4	Total
Empathy	0	2	14	9	25	7	2	0	0	9
Defining/Point of View (POV) Development	0	4	16	5	25	7	1	0	0	8
Idea Creation	0	0	12	13	25	8	10	4	0	22
Prototyping	0	0	17	8	25	15	8	1	0	24
Testing	0	8	11	6	25	4	4	0	0	8

When Table 14 is analyzed, it is seen that the majority of the students in the experimental group have exhibited DT behaviors at Level 3, and the majority of the students in the control group exhibited DT behaviors at Level 1. Accordingly, it can be said that the students in the experimental group exhibited higher levels of DT behaviors as a result of the activities applied.

### *Findings of the Fifth Sub-Problem*

The documents related to the DT tasks performed during the experimental process (empathy map, POV development, and user feedback worksheets, note papers used in creating ideas) and the prototypes developed were analyzed using the DT Rubric developed by the researcher and how these documents reflect the DT process of was analyzed in the fifth sub-problem of the study. In DT tasks, students worked in 5 groups of 5 people. The mean of all groups' scores of the DT rubric in the first DT task was 19.2 out of 20, 18.8 in the second DT task, 19 in the third DT task, 19.4 in the fourth DT task and 19.6 in the fifth DT task. When all DT tasks were examined together, the mean of the students' scores in the experimental group from the DT rubric was 19.2. It can be said that the mean of the students' scores in the experimental group from the DT rubric were high.

### **Discussion and Conclusion**

In this section, conclusions and findings for each research question are discussed by referring to relevant literature and studies conducted.

1. Research Question: *Does the instructional design affect the computational thinking, creative thinking and programming self-efficacy scores of gifted and talented students?*

As a result of the study, it is shown that instructional design has effects on developing computational thinking skills of gifted and talented students. In literature, there hasn't been any study available that searches the effects of programming-centered teaching strategies on gifted and talented students' computational thinking skills. However, Kim et al. (2013) and Wang, Huang, and Hwang (2014) concluded that teaching designs with a programming-centered approach affected the problem-solving skills of gifted students. It can be said that the results of previous research match with this research results, considering that problem-solving skill is an element of computational thinking skill (ISTE, 2016) and explains 24% of computational thinking (Saritepeci, 2017). Ozcelik and Akgunduz (2018) concluded that at the end of application instructional design that focuses on STEM education, skills of gifted and talented students such as creativity, critical thinking, and cooperation have been improved. Similarly, Ayverdi (2018) stated that teaching design was effective in scientific creativity and the scientific process skills of gifted and talented students. These studies are in parallel to the results of this study in terms of showing that skills covered by computational thinking can be improved by instructional designs in gifted and talented students whereas they have differences in terms of focusing on programming teaching.

Researches are available in the literature presenting evidence that students with a normal level of ability can enhance their computational thinking skills with programming-based instructional designs (Bers et al., 2014; Kim & Kim, 2016; Saez-Lopez et al., 2016). The results of previous studies and this study are aligned. However, it was indicated in the studies carried out by Atman et al. (2018) and Weese and Feldhausen (2017) that instructional designs were not adequately effective in the development of computational thinking skills. In previous studies, the attempt to develop computational thinking was executed by using block-based programming pedagogy only. In this study, block-based, text-based, and physical programming pedagogies and design thinking have been combined, and instructional design has been executed co-operatively, and it has been observed that the computational thinking skills were improved.

It has been seen that instructional design has a high level of effect (effect size is at a high level) on the creative thinking skills of gifted and talented students. In literature, there isn't any study available showing that programming-based instructional designs are developing gifted and talented students' creative thinking skills. Nonetheless, Ayverdi (2018) and Ozcelik and Akgunduz (2018) studies indicate that teaching designs focused on STEM are effective in gifted and talented students' creative thinking skills. The outcomes of previous studies coincide, in this respect, with the findings of this study. It has been stated that creative thinking skills can be cultivated with programming-based instructional designs for students with a normal skill level (Ataman Uslu et al., 2018; Clements & Gullo, 1984; Kim & Kim, 2016; Kobsiripat, 2015; Pardemean et al., 2011; Park et al., 2015). Design thinking activities in instructional design are claimed to have a significant impact on the high degree of change in gifted and talented students' creativity. Students use their creative thinking skills in every stage of the design thinking process (Henriksen et al., 2017). There are also studies in the literature that provide indications of the improvement of creative thinking skills as a result of the application of design thinking practices (Carroll et al., 2010; Koh et al., 2015; Noel & Liub, 2017), and past results and current results are mutually supportive.

Instructional design has not been effective in programming the self-efficacy of gifted and talented students. Although it does not concentrate on gifted and talented students, similar findings have been found through the execution of the programming-based instructional designs (Davidsson et al., 2010; Korkmaz, 2016; Ortiz, Chiluiza & Valcke, 2017). It was concluded in the studies carried out by Gunbatar and Karalar (2018) and Soykan and Kabul (2018) that instructional designs increased students' programming self-efficacy. The findings of these are not consistent with the present study. High pre-test scores of gifted and talented students are thought to have been effective in this condition. Furthermore, a deficiency of at least one of the resources that encourage self-efficacy or a deficiency related to one of many psychological factors such as confidence, motivation, desire, and psychological resilience may have influenced programming self-efficacy.

2. Research Question: *What are the views of gifted and talented students about computational thinking before and after the implementation of the instructional design?*

Regarding interviews with four students in the experimental and control groups before and after the experimental process, it was demonstrated that the students' computational thinking skills in the experimental group where the instructional design was implemented, improved more than the students in the control group. Following the research, the students in the experimental group chose to speak more about the computational concepts and applications, after the experimental process than the students in the control group. Moreover, after the experimental process, the lack of information on computational concepts was not present. Studies in literature indicate that teaching designs have positive effect on students' use of computational concepts (Faber, Wierdsma, Doornbos, van der Ven & de Vette, 2017; Constantinou & Ioannou, 2018; Rodrigez, 2017) and the implementations they apply when they produce something by programming (Atmatzidou & Demetriadis, 2016; García-Valcarcel et al., 2019; Sullivan et al., 2017). The results of previous studies are in line with the results of this study.

During the pre- and post-interviews, the students in the experimental and control groups claimed an equal number of views about their computational perspectives. All groups, however, gained computational perspectives. Likewise, it was stated that, as mentioned in the studies of Burke (2012), Kahn et al. (2011), and Lin and Liu (2012), students acquired computing perspectives through programming teaching. The students' recent interviews in the experimental group may have stemmed from the same number of computational perspectives and their emphasis on cognitive achievements in the instructional design.

3. Research Question: *What are the views of gifted and talented students about design thinking before and after the implementation of instructional design?*

The students in the experimental group mentioned the design thinking process, prototyping tools, and designer features following the experimental process. Despite their views before the experimental process, the students mentioned that after the experimental process they learned the content of the course, in the defining stage they established POV statements, also that the process was iterative, and that they enjoyed the process of design thinking. No study applies design thinking to gifted and talented students in the literature. However, the engineering design cycle whose stages are similar to the design thinking process was used in the works of Ayverdi (2018) and Ozcelik and Akgunduz (2018), and as a consequence of the teaching process, it was stated that the gifted and talented students learned the course material, also the design process and enjoyed the process. The preceding research results and the results of this research are mutually supportive. Similarly, it has been determined in the literature that students with the normal level of abilities have learned the content of the course (Carroll et al., 2010; Carroll, 2015; Kwek, 2011; Painter, 2018) and enjoyed the design process (Dukes & Koch, 2012). Another result obtained in both the pre-interviews and the post-interviews (for both groups) is linked to the fact that during the design process there may be disputes between group members and the difficulty of working with the team may be observed. While working together, disagreements between team members were observed in the works of Santos Ordonez, Gonzalez Lema and Mino Puga (2017) and Retna (2016). We are known to encounter challenges when interacting with the team because of the leadership personality trait that resides in almost all of the gifted and talented students. Unlike the preliminary interviews, the experimental group students discussed a successful designer's characteristics to empathize, dignify the process of thinking, collaboration with the team, and appreciate the thoughts of others. Design thinking activities are considered to be useful in instructional design.

While the students in the control group shared their perspectives on the stages of empathy, brainstorming, and prototyping related to the design thinking process before the experimental process; in the post interviews, they also expressed opinions on the processes of brainstorming, prototyping, and testing. This may be due to the use of similar processes existing in software development methods, and the use of an engineering design cycle in STEM courses that have similar points to the gifted and talented students' design thinking process.

4. Research Question: *What are the gifted and talented students' state of using computational thinking and design thinking in terms of teacher observations during the experimental process?*

The findings obtained from the experimental process-related teacher observations show that the computational thinking skills of the students in the experimental group have advanced more than those of the control group. Experimental group students have displayed more computational behavior than control group students. The behaviors in the observation form of computational thinking are related to the applications that the learners use when they generate something through programming. These applications are experimenting and iterating during the design, testing and debugging of the code, reusing and remixing the code of others, abstracting, and modularizing (Brennan & Resnick, 2012). It is assumed that block-based, text-based, and physical programming activities and programming process enrichment with design thinking in instructional design support students demonstrate computational behavior.



According to teacher observations, it is recognized that most students in the experimental group display Level 3 behaviors while the majority of students in the control group demonstrate Level 1 behaviors. Students in the experimental group demonstrated greater levels of behavior related to their skills in design thinking than the control group. In the literature, there is no study examining the impact of instructional designs that were introduced to gifted students on design thinking skills. However, Ayverdi (2018) indicated that the teaching design developed by complying with the general teaching design model has been effective in gifted and talented students' engineering skills (stages similar to the DT process). The claim made in the study of Ayverdi (2018), and the conclusion reached in this study is mutually supportive. In the work of Duman and Kayali (2017), it is observed that teaching activities concentrating on the design thinking process were effective in the development of secondary school students' design thinking skills. The bottom line reached by Aflatoony et al. (2018) shows that high school students develop their design thinking skills to a certain extent through the design thinking program established. The outcomes of this study coincide with the data obtained by Duman and Kayali (2017) and Aflatoony et al. (2018).

5. Research Question: *How do design thinking worksheets, note sheets used in the idea-making process, and prototypes developed by students reflect the design thinking process of gifted and talented students?*

The scores of the students in the experimental group regarding the design thinking tasks were measured by the rubric of design thinking, and it was grasped that the students' average score was 19.2 out of 20 for the five design tasks given. This result shows that the average scores of the students obtained from the design thinking rubric are quite high. All the stages of the process were meticulously applied by the students who participated in the design thinking process, and during this process, they filled out their empathy map, POV statements, user feedback templates, wrote their ideas on note papers in the stages of brainstorming and launched their ideas using block-based, text-based, and physical programming instruments. This is an indication that students enjoy the process of design thinking, and are motivated by it.

In summary, as a result of the holistic analysis of quantitative and qualitative data, it was concluded that the instructional design applied to the experimental group was successful in improving the computational, creative thinking and design thinking skills of gifted and talented students, while it was not effective in programming self-efficacy of the students.

### Limitations and Suggestions

Only pre-test and post-test measurements could be taken regarding the dependent variables in the study, and the retention test could not be performed. In the following researches, the persistence effect of instructional design can be examined. The instructional design developed within the scope of the research was implemented at BILSEM by using the facilities of the institution and project support. The instructional design developed within the scope of this research can be applied in institutions with similar infrastructure or teachers can develop their designs. The generalizability of the research results can be ensured when instructional design developed for gifted and talented students in institutions with similar infrastructure is applied. Computational thinking skills can be evaluated with different assessment approaches and tools available in the literature. The answer to the question of what is in the minds of students can be sought during the implementation of instructional design. Other sources (verbal persuasion, persuasive feedback, etc.) that affect programming self-efficacy may be introduced in the implementation process of instructional design. The effect of instructional design on the success of programming can be investigated. Creative thinking skills can be examined through context-based creativity tests to be developed in the field of computer science. Student roles (participation, teamwork, collaboration) can be examined in more detail through observations made during the DT activities. The impact of using DT as a method on learning 21st-century skills, social competences, motivation, interest-will, attitude, and course content can be extensively studied.

### References

- Aflatoony, L., Wakkary, R., & Neustaedter, C. (2018). Becoming a design thinker: assessing the learning process of students in a secondary level design thinking course. *The International Journal of Art & Design Education*, 37(3), 438-453.
- Akcaoglu, M., & Koehler, M. J. (2014). Cognitive outcomes from the game-design and learning (GDL) after-school program. *Computers & Education*, 75, 72-81.
- Askar, P., & Davenport, D. (2009). An investigation of factors related to self-efficacy for java programming. *The Turkish Online Journal of Educational Technology - TOJET January*, 8(1), 26-32.
- Atman Uslu, N., Mumcu, F., & Egin, F. (2018). Gorsel programlama etkinliklerinin ortaokul ogrencilerinin bilgi-islemselel dusunme becerilerine etkisi [The effect of visual programming activities on secondary school students' computational thinking skills]. *Ege Journal of Educational Technologies/Ege Egitim Teknolojileri Dergisi*, 2(1), 19-31.

- 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.
- Ayverdi, L. (2018). Usage of technology, engineering and mathematics in science education for gifted students: STEM approach (Unpublished doctoral dissertation). University of Balikesir, Balikesir, Turkey.
- Baer, J., & Kaufman, J. C. (2012). *Being creative inside and outside the classroom, how to boost your students' creativity and your own*. Sense Publishers.
- Balanskat, A., & Engelhardt, K. (2015). *Computing our future computer programming and coding priorities, school curricula and initiatives across Europe*. [https://www.dzs.cz/file/3394/computing-our-future\\_final-pdf](https://www.dzs.cz/file/3394/computing-our-future_final-pdf)
- Balanskat, A., Engelhardt, K., & Ferrari, A. (2017). *The integration of computational thinking across school curricula in Europe*. [https://publications.jrc.ec.europa.eu/repository/bitstream/JRC104188/jrc104188\\_computhinkreport.pdf](https://publications.jrc.ec.europa.eu/repository/bitstream/JRC104188/jrc104188_computhinkreport.pdf)
- Begosso, L., & da Silva, P. (2013, October). *Teaching computer programming: a practical review*. Paper presented at IEEE Frontiers in Education Conference (FIE), Oklahoma City, USA.
- Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers and Education*, 72, 145-157.
- Brennan, K., & Resnick, M. (2012, April). *Using artifact-based interviews to study the development of computational thinking in interactive media design*. Paper presented at Annual American Educational Research Association Meeting, Vancouver, BC, Canada.
- Bootcamp Bootleg D. School (2011). *Design thinking*. Institute of Design at Stanford, <http://longevity3.stanford.edu/designchallenge2015/files/2013/09/Bootleg.pdf>
- Burke, Q. (2012). The Markings of a New Pencil: Introducing programming-as-writing in the middle school classroom. *The National Association for Media Literacy Education's Journal of Media Literacy Education*, 4(2), 121-135.
- Buyukozturk, S. (2014). *Deneysel desenler ontest-sontest kontrol grubu desen ve veri analizi* [Experimental designs pre-test and post-test control group pattern and data analysis]. Pegem Akademi.
- Carroll, M., Goldman, S., Britos, L., Koh, J., Royalty, A., & Hornstein, M. (2010). Destination, imagination and the fires within: design thinking in a middle school classroom. *International Journal of Art & Design Education*, 29(1), 37–53.
- Carroll, M. (2014). Shoot for the moon! The mentors and the middle schoolers explore the intersection of design thinking and STEM. *Journal of Pre-College Engineering Education Research*, 4(1), 14-30.
- Calao L. A., Moreno-Leon, J., Correa, H. E., & Robles, G. (2015). Developing mathematical thinking with scratch. an experiment with 6th grade students. In G. Conole, T. Klobucar, C. Rensing, J. Konert, & E. Lavoue (Eds.), *Design for teaching and learning in a networked world* (pp. 17-27). Springer International Publishing.
- Creswell, J. W., & Plano Clark, V. L. (2011). *Designing and conducting mixed methods research* (2nd ed.). Thousand Oaks, CA: Sage.
- Clements, D. H., & Gullo, D. F. (1984). Effects of computer programming on young children's cognition. *Journal of Educational Psychology*, 76(6), 1051-1058.
- Cakiroglu, U, Sari, E., & Akkan, Y. (2011, September,). *The view of the teachers about the contribution of teaching programming to the gifted students in the problem solving*. Paper presented at the 5th International Computer and Instructional Technologies Symposium, Elazig, Turkey.
- Constantinou, V., & Ioannou, A. (2018, September). *Development of computational thinking skills through educational robotics*. Paper presented at 13th European Conference on Technology Enhanced Learning, Leeds, UK.
- Colluoglu-Gulen, O. (2014). Analysis of gifted students' interest areas using data mining techniques (Unpublished doctoral dissertation). Gazi University, Ankara, Turkey.
- Davidson, K., Larzon, L., & Ljunggren, K. (2010). *Self-efficacy in programming among STS students*. Retrieved from <http://www.it.uu.se>.
- Dukes, C., & Koch, K. (2012). Crafting a delightful experience: teaching interaction design to teens. *Interactions*, 19(2), 46-50.
- Duman, B., & Kayali, D. (2017). Teknopedagogjik ogretme yaklasiminin tasari odakli dusunme becerilerine etkisi [The effect of techno-pedagogical teaching approach on design thinking skills]. In B. Akkoyunlu, A. Isman, & H. F. Odabasi (Eds.), *Egitimde Teknoloji Okumalari 2018* [Technology Readings in Education 2018] (pp.176-184). TOJET.
- Durak, H. (2016). Design and development of an instructional program for teaching programming process to gifted students (Unpublished doctoral dissertation). Gazi University, Ankara, Turkey.

- Faber, H. H., Wierdsma, M. D., Doornbos, R. P., van der Ven, J. S., & de Vette, K. (2017). Teaching computational thinking to primary school students via unplugged programming lessons. *Journal of the European Teacher Education Network, 12*, 13-24.
- Garcia-Valcarcel Munoz-Repiso, A., & Caballero-Gonzalez, Y. A. (2019). Robotics to develop computational thinking in early childhood education. *Media Education Research Journal, 59*(17), 63-72.
- Geckil, A. (2012). Evaluation of laboratory qualifications and applications in science and art centers (BILSEM). (Unpublished master's thesis). Gazi University, Ankara, Turkey.
- Grover, S., & Pea, R. (2013, March). *Using a discourse-intensive pedagogy and android's app inventor for introducing computational concepts to middle school students*. Paper presented at SIGCSE'13, Colorado, USA.
- Gulbahar, Y. (2018). Bilgi islemsel dusunme ve programlama konusunda degisim ve donusumler [Changes and transformations in computational thinking and programming]. In Y. Gulbahar (Ed.), *Bilgi islemsel dusunmeden programlamaya* [From computational thinking to programming] (pp.395-411). Pegem Akademi.
- Gunbatar, M. S., & Karalar, H. (2018). Gender differences in middle school students' attitudes and self-efficacy perceptions towards mBlock programming. *European Journal of Educational Research, 7*(4), 925-933.
- Henriksen, D., Richardson, C., & Mehta, R. (2017). Design thinking: a creative approach to educational problems of practice. *Thinking Skills and Creativity, 26*, 140-153.
- Hongwarittorn, N., & Krairit, D. (2010, April). *Effects of program visualization on students' performance and attitudes towards java programming*. Paper presented at the International conference on Computing, Communication and Control Technologies, Orlando, Florida.
- ISTE (2016). *ISTE standarts for students*. Iste Standards for Students. <https://www.iste.org/standards/for-students>
- Jonassen, D. H. (2000). *Computers as mind tools for schools: engaging critical thinking*. Prentice Hall.
- Kahn, K., Sendova, E., Sacristan, A. I., & Noss, R. (2011). Young students exploring cardinality by constructing infinite processes. *Technology, Knowledge and Learning, 16*(1), 3-34.
- Kalelioglu, F., & Gulbahar, Y. (2014). The effects of teaching programming via Scratch on problem solving skills: a discussion from learners' perspective. *Informatics in Education, 13* (1), 33-50.
- Kalelioglu, F. (2018). Turkiye'de programlama ogretimi [Programming teaching in Turkey]. In Y. Gulbahar, & H. Karal (Eds.), *Kuramdan uygulamaya programlama ogretimi* [Theory-to-practice programming teaching] (pp.68-89). Pegem Akademi.
- Kandemir, C. M. (2018). Metin tabanlı programlama. [Text-based programming]. In Y. Gulbahar, & H. Karal (Eds.), *Kuramdan uygulamaya programlama ogretimi* [Theory-to-practice programming teaching] (pp.299-336). Pegem Akademi.
- Kaplan-Sayi, A. (2013). The effect of differentiated foreign language instruction on gifted students' achievement, critical thinking and creativity. (Unpublished doctoral dissertation). University of Istanbul, Istanbul, Turkey.
- Karaman, S., & Kursun, E. (2018). Programlama ogretiminde degerlendirme yaklasimlari [Evaluation approaches in programming teaching]. In Y. Gulbahar, & H. Karal (Eds.), *Kuramdan uygulamaya programlama ogretimi* [Theory-to-practice programming teaching] (pp.434-477). Pegem Akademi.
- Kelley, T., & Kelley, D. (2013). *Creative confidence: unleashing the creative potential within us all*. Random House Company.
- Kert, S. B. (2018a). Bilgisayar bilimi egitimine giris [Introduction to computer science education]. In Y. Gulbahar (Ed.), *Bilgi islemsel dusunmeden programlamaya* [From computational thinking to programming] (pp. 1-20). Pegem Akademi.
- Kert, S. B. (2018b). Programlama ogretimi icin pedagogik yaklasimlar [Pedagogical approaches for programming teaching]. In Y. Gulbahar, & H. Karal (Eds.), *Kuramdan uygulamaya programlama ogretimi* [Theory-to-practice programming teaching] (pp.93-130). Pegem Akademi.
- Keskin, S. (2006). Gifted and talented and attitude towards computer and computer course (Unpublished master's thesis). University of Balikesir, Balikesir, Turkey.
- Kircaburun, K, Bastug, I., and Bahtiyar, M. (2017). Modeling the psychological factors affecting computer programming self-efficacy. *Anatolian Journal of Educational Leadership and Instruction, 5*(1), 17-27.
- Kim, S., Chung, K., & Yu, H. (2013). Enhancing digital fluency through a training program for creative problem solving using computer programming. *The Journal of Creative Behavior, 47*, 171-199.

- Kim, Y. M., & Kim, J. H. (2016). Application of a software education program developed to improve computational thinking in elementary school girls. *Indian Journal of Science and Technology*, 9(44), 1-9.
- Kobsiripat, W. (2015). Effects of the media to promote the Scratch programming capabilities creativity of elementary school students. *Procedia-Social and Behavioral Sciences*, 174, 227-232.
- Koh, J. H. L., Chai, C. S., Wong, B., & Hong, H. (2015). *Design thinking for education: conceptions and applications in teaching and learning*. Springer.
- Kong, S. C. (2017). Development and validation of a programming self-efficacy scale for senior primary school learners. In S. C. Kong, J. Sheldon, & K. Y. Li (Eds.), *Proceedings of the International Conference on Computational Thinking Education* (pp. 97-102). The Education University of Hong Kong.
- Kong, S. C. (2019). Components and methods of evaluating computational thinking for fostering creative problem-solvers in senior primary school education. In: Kong SC., & Abelson H. (Eds.), *Computational Thinking Education* (pp.119-141). Springer.
- Koo, T. K., & Li, M. Y. (2016). A guideline of selecting and reporting intraclass correlation coefficients for reliability research, *Journal of Chiropractic Medicine*, 15, 155-163.
- Korkmaz, O., Cakir, R., & Ozden, M. Y. (2015). Bilgisayarca dusunme beceri duzeyleri olceginin ortaokul duzeyine uyarlanmasi [Computational thinking levels scale adaptation for secondary school level]. *Gazi Journal of Education Sciences/Gazi Egitim Bilimleri Dergisi*, 1(2), 67-86.
- Korkmaz, O. (2016). The effect of Lego mindstorms EV3 based design activities on students' attitudes towards learning computer programming, self-efficacy beliefs and levels of academic achievement. *Baltic J. Modern Computing*, 4(4), 994-1007.
- Krathwohl, D. R., Bloom, B. S., & Masia, B. B. (1964). *Taxonomy of educational objectives: the classification of educational goals. handbook II: affective domain*. McKay.
- Kukul, V., Gokcearslan, S., & Gunbatar, M. S. (2017). Computer programming self-efficacy scale for secondary school students: Development, validation and reliability. *Educational Technology Theory and Practice*, 7(1), 158-179.
- Landis, J. R., & Koch, G. G. (1977). The Measurement of observer agreement for categorical data. *International Biometric Society*, 33(1), 159-174.
- Lin, J. M.-C., & Liu, S.-F. (2012). An investigation into parent-child collaboration in learning computer programming. *Educational Technology & Society*, 15(1), 162-173.
- Lockwood, J., & Mooney, A. (2017). *Computational thinking in education: Where does it fit? a systematic literary review*. Cornell University. <https://arxiv.org/abs/1703.07659>
- Lor, R. R. (2017, May). *Design thinking in education: a critical review of literature*. Paper presented at Asian Conference on Education & Psychology, Bangkok, Thailand.
- Lye, S. Z., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, 41, 51-61.
- McGuinness, C., & O'Hare, L. (2012). Introduction to the special issue: new perspectives on developing and assessing thinking. *Thinking Skills and Creativity*, 7(2), 75-77.
- Miles, M. B., & Huberman, A. M. (1994). *Qualitative data analysis: an expanded sourcebook*. SAGE Publications.
- Ministry of National Education (2016). *Science and art centers directive*. Ministry of National Education. [https://orgm.meb.gov.tr/meb\\_iys\\_dosyalar/2016\\_10/07031350\\_BILSEM\\_yonergesi.pdf](https://orgm.meb.gov.tr/meb_iys_dosyalar/2016_10/07031350_BILSEM_yonergesi.pdf)
- Ministry of National Education (2017). *Science and art center information technologies and software course framework program*. Retrieved from <http://BILSEM.meb.gov.tr/login.aspx>
- Morrison, G. R., Ross, S. M., & Kemp, J. E. (2001). *Designing effective instruction*. John Wiley.
- Noel, L., & Liub, T. (2017). Using design thinking to create a new education paradigm for elementary level children for higher student engagement and success. *Design and Technology Education: An International Journal*, 1(22), 501-512.
- Orhon, G. (2014). *Yaratıcılık, norofizyolojik, felsefi ve egitsel temeller* [Creativity, neurophysiological, philosophical, and educational foundations]. Pegem A.
- Ortiz, M., Chiluiza, K., & Valcke, M. (2017, October). *Gamification in computer programming: effects on learning, engagement, self-efficacy and intrinsic motivation*. Paper presented at 11th European Conference on Games Based Learning, Graz, Austria.

- Ongo, S., & Sozel, H. K. (2018). Ustun yeteneklilerin egitiminde teknoloji kullanimi [The use of technology in gifted education]. In H. F. Odabasi (Ed.), *Ozel egitim ve egitim teknolojisi* [Special education and educational technology] (pp.91-114). Pegem Akademi.
- Ozcelik, A., & Akgunduz, D. (2018). Ustun/ozel yetenekli ogrencilerle yapilan okul disi STEM egitiminin degerlendirilmesi [Evaluation of gifted/talented students' out-of-school STEM education]. *Trakya Universitesi Egitim Fakultesi Dergisi*, 8(2), 334-351.
- Painter, D. (2018). Using design thinking in mathematics for middle school students: a multiple case study of teacher perspectives. (Unpublished master thesis). Concordia University, Portland, USA.
- Park, I., Kim, D., Oh, J., Jang, Y., & Lim, K. (2015). Learning effects of pedagogical robots with programming in elementary school environments in Korea. *Indian Journal of Science and Technology*, 8(26), 1-5.
- Pardamean, B., Evelin, E., & Honni, H. (2011, December). *The effect of Logo programming language for creativity and problem solving*. Paper presented at 10th WSEAS International Conference on EActivities, Jakarta, Indonesia.
- Plucker, J. A., & Callahan C. M. (2014). Research on giftedness and gifted education. *Exceptional Children*, 80(4), 390-406.
- Ramalingam, V., & Wiedenbeck, S. (1998). Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self efficacy. *Journal of Educational Computing Research*, 19(4), 367-381.
- Rauth, I., Koppen, E., Jobst, B., & Meinel, C. (November, 2010). Design thinking: an educational model towards creative confidence. Paper presented at 1st International Conference on Design Creativity, Kobe, Japan.
- Repenning, R., Webb, D.C., Koh, K.H., Nickerson, N., Miller, B., Brand, S. B., & Repenning, N. (2015). Scalable game design: a strategy to bring systemic computer science education to schools through game design and simulation creation. *Trantions on Computing Education*, 15(2), 1-31.
- Retna, K. S. (2016). Thinking about design thinking: a study of teacher experiences. *Asia Pacific Journal of Education*, 36(1), 5-19.
- Rodriguez, B. R. (2017). Assessing computational thinking in computer science unplugged activities. (Unpublished master's thesis). Colorado School of Mines, USA.
- Roman-Gonzalez, M., Moreno-Leon, J., & Robles, G. (2019). Combining assessment tools for a comprehensive evaluation of computational thinking interventions. In S. C. Kog, & H. Abelson (Eds.), *Computational thinking education* (pp.79-98). Springer Open Access.
- Romero, M., Laferriere, T., & Power, T. M. (2016). The move is on! from the passive multimedia learner to the engaged co-creator. *eLearn Magazine*, 3, 1-6.
- Romero, M., Lepage, A., & Lille, B. (2017). Computational thinking development through creative programming in higher education. *International Journal of Educational Technology in Higher Education*, 14(42), 1-15.
- Saez-Lopez, J.M., Roman-Gonzalez, M., & Vazquez-Cano, E. (2016). Visual programming languages integrated across the curriculum in elementary school: a two year case study using scratch in five schools. *Computers & Education*, 97, 129-141.
- Santos Ordenez, A., Gonzalez Lema, C., & Mino Puga, M. F. (2017, July). *Design thinking as a methodology for solving problems: contributions from academia to society*. Paper presented at 15th LACCEI International Multi-Conference for Engineering, Education, and Technology, Boca Raton FL, USA.
- Saritepeci, M. (2017, October). *Computational thinking skill level in secondary education in terms of various variables*. Paper presented at 5th International Instructional Technologies & Teacher Education Symposium, Izmir, Turkey.
- Shin, S., Park, P., & Bae, Y. (2013). The effects of an information-technology gifted program on friendship using scratch programming language and clutter. *International Journal of Computer and Communication Engineering*, 2(3), 246-249.
- Siegle, D. (2004). Identifying students with gifts and talents in technology. *Gifted Child Today*, 27(4), 30-34.
- Simsek, A. (2017). *Ogretim tasarimi* [Instructional design]. Nobel Academic Publication.
- Soykan, F., & Kanbul, S. (2018). Analysing K12 Students' self-efficacy regarding coding education, *TEM Journal*, 7(1), 182-187.
- Starko, A. J. (2014). *Creativity in the classroom schools of curious delight*. Routledge.

- Tyler-Wood, T. L., Mortenson, M., Putney, D., & Cass, M. A. (2000). An effective mathematics and science curriculum option for secondary gifted education. *Roeper Review*, 22(4), 266-269.
- VanTassel-Baska, J., Bass, G., Ries, R., Poland, D., & Avery, L. D. (1998). A national study of science curriculum effectiveness with high ability students. *Gifted Child Quarterly*, 42(4), 200-211.
- Wang, H. Y., Huang, I., & Hwang, G. J. (2014, August). Effects of an integrated Scratch and project-based learning approach on the learning achievements of gifted students in computer courses. In *2014 IIAI 3rd International Conference on Advanced Applied Informatics* (pp. 382-387). IEEE.
- Weese, J. L., & Feldhausen, R. (2017, June). *STEM outreach: assessing computational thinking and problem solving*. Paper presented at ASEE Annual Conference & Exposition, Columbus, Ohio.
- Wing, J. M. (2011). *Computational thinking: What and why*. Carnegie Mellon University. <http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>
- Yildiz-Durak, Karaoglan Yilmaz, F. G., & Yilmaz, R. (2019). Computational thinking, programming self-efficacy, problem solving and experiences in the programming process conducted with robotic activities. *Contemporary Educational Technology*, 10(2), 173-197.

Annexes

Annex-1. The Detailed Information about the Activities in the Instructional Design

Units	Activities	Time	Audio-Visual and Interactive Materials
Unit 1: Design Thinking	Before School (DT1)	120'	Design Thinking (DT) digital presentation, Actionbound mobile app, mobile phone and / or tablet computer, Team Gauge Game, other materials (A4 papers, sticky note papers, colored pencils, prototyping materials)
	Livable Place (DT2)	120'	Design Thinking (DT) digital presentation, Empathy Map Sample, Empathy Map Template, Perspective Development Worksheet, User Feedback Worksheet, Videos, Wordart and Onlineclassroomwatch Web 2.0 Tools, Tool Gauge Game, other materials (A4 papers, sticky note papers, colored pencils, prototyping materials)
Unit 2: Block-Based Programming	Archery	120'	Mentimeter Web 2.0 tool, Life Ear Game (Carpet Game), Block Based Programming Environment (Scratch or Mblock).
	Jet Ski	80'	VR Cardboard, VR Jet Ski Application, A Nonsense Story Worksheet, Block Based Programming Environment (Scratch or Mblock).
	Memory Games	80'	Crazy Comparisons Worksheet, Gonoodle and Dr. Scratch Web 2.0 tool, Block Based Programming Environment (Scratch or Mblock).
	Catch the Balls	120'	Arrays Worksheet, Topic Game (Carpet Game), Block Based Programming Environment (Scratch or Mblock)
Unit 3: Text-Based Programming	Design Thinking Project Development 1	240'	Design Thinking (DT) digital presentation, Empathy Map Template, Perspective Development Worksheet, User Feedback Worksheet, onlineclassroomwatch, Kahoot web 2.0 tools, Tool Gauge Game, other materials (A4 papers, sticky note papers, colored pencils), Block Based Programming Environment (Scratch or Mblock)
	Rock Paper Scissors	120'	Python text-based programming environment or computer connected to the Internet (if Trinket will be used), Python flash cards
	Turtle Geometry	80'	Python text-based programming environment or internet-connected computer (if Trinket will be used), Python flash cards, drawing cards, A4 paper and pens.
	Choose Your Team	80'	Python text-based programming environment or computer connected to the Internet (if Trinket will be used), A4 paper and colored pencils.
	You Have A Secret Message	80'	Python text-based programming environment or computer connected to the Internet (if Trinket will be used), A4 paper and colored pencils.
Unit 4: Physical Programming	Design Thinking Project Development 2	240'	Design Thinking (DT) digital presentation, Empathy Map Template, Perspective Development Worksheet, User Feedback Worksheet, onlineclassroomwatch, Kahoot web 2.0 tools, Tool Gauge Game, other materials (A4 papers, sticky note papers, colored pencils), Text Based Programming Environment (Python)
	Smart Systems	120'	Circuit Schemes Worksheet, LDR sensor, Ultrasound Distance Sensor, Color and White Leds, Servo Motor, Buzzer, Jumper Cable, Resistors, Micro Controller Types (Arduino Uno, Nano, Mega, Lillypad and Raspberry PI 3) and USB Cable, Breadboard, Cardboard, Paint, Toy Cars, Prototyping Materials, Block Based Robot Programming Environment (Mblock)
	Agricultural Automation System	120'	Arduino Uno, Lm35 Temperature Sensor, Soil Moisture Sensor (Hygrometer), Liquid Level / Rain Sensor, 16x2 Lcd I2c, Buzzer, Red Led, Resistors, Mblock Lcd I2c Pack Attachment, Large Container, Soil, Water in Pet Bottle, Various Prototyping Materials, Kahoot Web 2.0 Tool, Block Based Robot Programming Environment (Mblock)
	Design Thinking Project Development 3	240'	Design Thinking (DT) digital presentation, Empathy Map Template, Perspective Development Worksheet, User Feedback Worksheet, Videos, Onlineclassroomwatch, Kahoot web 2.0 tools, Arduino Uno Ultimate Set, Tool Gauge Game, other materials (A4 papers, sticky note papers, colored pencils), Block Based Robot Programming Environment (Mblock)
	Mind Storm	160'	Lego Mindstorm EV3 Educational Kits, Flash cards, Task 1 and 2 Runways, Colorful Electrical Bands, Various Materials That Can Be Used As Obstacles and Loads, Block Based Robot Programming Environment (Legomindstorms Education EV3 Teacher Edition)
	Advanced Models with Variables	80'	Lego Mindstorm EV3 Educational Kits, Mission Track, Black Electrical Bands, Block Based Robot Programming Environment (Legomindstorms Education EV3 Teacher Edition)
Unit 5: Mobil Programming	Design Tinking Project Development 4	240'	Design Thinking (DT) Digital Presentation, Empathy Map Template, Perspective Development Worksheet, User Feedback Worksheet, Videos, Onlineclassroomwatch, Kahoot web 2.0 tools, Tool Gauge Game, Lego Mindstorm EV3 Educational Kits and Add-on Sets, Block Based Robot Programming Environment (Legomindstorms Education EV3 Teacher Edition)
	I am writing a mobile application	160'	Mobile Phone with Android Operating System, MIT AI2 Companion and Xing Barcode Scanner Applications, Flash Cards, Block Based Mobile Programming Environment (App Inventor 2)
	Mole Hunt	120'	Mobile Phone with Android Operating System, MIT AI2 Companion Application, Block Based Mobile Programming Environment (App Inventor 2)
Unit 5: Mobil Programming	Design Thinking Project Development 5	240'	Design Thinking (DT) Digital Presentation, Empathy Map Template, Perspective Development Worksheet, User Feedback Worksheet, Videos, Onlineclassroomwatch, Kahoot Web 2.0 Tools, Tool Gauge Game, Block Based Mobile Programming Environment (App Inventor 2)

Annex-2. Empathy Map Template, POV Template, User Feedback Template and DT Rubric

Empathy Map Template

What They Said / Done	My Thoughts / Feelings
Interviewer's observations that are seen and heard stated in this section. The sentences that come out of the interviewee's mouth are included in the quotes without being changed.	Based on the data in the what they said / done section, deductions are made about the user and assumptions are noted.
...	...

POV Template

.....'s, (user's name)

..... (verb) needs to find a way / method for that,

Because.....

(insight)

User Feedback Template

What worked out?	What did not work out?
...	...
What should be improved?	What has surprised us?
...	.....

Design Thinking Rubric

Criteria	Levels				Score
	1 (Weak)	2 (Moderate)	3 (Competent)	4 (Most Competent)	
Empathy	Interview questions are prepared proper to the design task and directed to user	The information gathered from interview noted randomly	The information gathered from the interview noted in empathy template by stating direct sentences coming from person's mouth	The information gathered from the interview was written down in the empathy template by including the sentences coming directly from the mouth of the person and the assumptions about the user stated.	
Point of View (POV)	One proper and clear POV statement has been written	Two clear POV statements have been written, partly properly	Three clear POV statements have been written almost properly	At least four clear POV statements have been written totally properly	
Brainstorming	0-5 ideas produced	5-15 ideas produced	15-25 ideas produced	More than 25 ideas produced	
Prototyping	No unnecessary things were included in the prototype, the design was partially interesting and visually successful.	No unnecessary things were included in the prototype, the design was very interesting and visually successful.	Easily used prototype with no unnecessary things and the design was very interesting and visually successful	Easily used prototype with no unnecessary things included and the design was very interesting and visually successful also enabled the user interaction	
Testing	Prototype is used by user and it is noted what has been worked out	Prototype is used by user and it is noted both what has been worked out and not	Prototype is used by user and it is noted both what has been worked out and not, with the points need to be improved	Prototype is used by user and it is noted both what has been worked out and not, with the points need to be improved and what have surprised design team	
<b>Total</b>					