

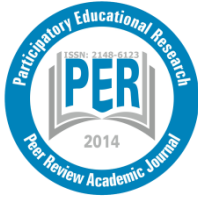
## PAPER DETAILS

TITLE: The effect of project-based arduino educational robot applications on students`  
computational thinking skills and their perception of Basic Stem skill levels

AUTHORS: Kübra KARAAHMETOGLU,Özgen KORKMAZ

PAGES: 1-14

ORIGINAL PDF URL: <https://dergipark.org.tr/tr/download/article-file/772664>



Participatory Educational Research (PER)  
Vol. 6(2), pp. 1-14, December, 2019  
Available online at <http://www.perjournal.com>  
ISSN: 2148-6123  
<http://dx.doi.org/10.17275/per.19.8.6.2>

## The effect of project-based arduino educational robot applications on students' computational thinking skills and their perception of Basic Stem skill levels

Kübra KARAAHMETOĞLU

*Amasya University, Science Institute, Department of Computer Education and Instructional Technology, Amasya, Turkey*

Özgen KORKMAZ\*

*Amasya University, Technology Faculty, Department of Computer Engineering, Amasya, Turkey*

<b>Article history</b>	<p>The aim of this study is to investigate the effects of project-based arduino educational robot applications on students' computational thinking skills and their perception of Stem skill levels. The study group consists of 6th grade students from 2 different secondary schools from Turkey. Within the scope of the research, classes were assigned to experimental and control groups neutrally. There are 15 students in the control group and 18 students in the experimental group. The experimental study continued for 11 weeks in both groups. In the experimental group, project-based arduino educational robot applications were applied in classes whereas in the control group, project development activities were carried out with block based programming tool. The research data were collected by using the computer-based skill level scale (<math>\alpha = 0.809</math>) and the Basic STEM Skill Levels scale (<math>\alpha = 0.940</math>). The mean, standard deviation, minimum, and maximum, Mann-Whitney U tests were used for independent measurements and the associated measurements were performed on the collected data. At the end of the research, it was determined that activities based on block based robotic programming tool did not have a significant effect on both students' total scores of Stem skills and scores related to factors, but when computational thinking skills were analyzed, it was found that they contributed significantly more than the total score and problem solving factor based on block based programming tool.</p>
<b>Received:</b> 24.04.2019	
<b>Received in revised form:</b> 14.05.2019	
<b>Accepted:</b> 15.05.2019	
<b>Key words:</b> Educational robot applications; Computational thinking; Perception of Stem skill levels	

### 1.Introduction

Existing teaching programs may be insufficient to meet the expectations of today's youth called the Z generation (Karabak and Güneş, 2013). In order to attract the attention of the Z generation, to support the traditional methods, and to enrich the educational environments, it has become obligatory to reorganize the teaching environments and teaching programs and this is also so as to adapt to the technological developments (Somyürek, 2014). Innovativeness race among the countries has increased even more with the acceleration of developments in the world and with the decreasing of resources. This situation also affected

\* ozgenkorkmaz@gmail.com

the education policies of the countries (Akgündüz et al., 2015). For this reason, many countries have to upbringing qualified and creative manpower who have a broad perspective, need to produce solutions to problems, use technology while creating their education strategies aimed at the next fifty or one hundred years' education strategies (Yenilmez and Balbağ, 2016). In this context, it is possible to say that technological developments have been used intensively in all areas of life and that the integration of technology to education has become a necessity in order for today's digital natives to use these tools correctly.

The Ministry of National Education of Turkey included the programming education which was previously present in the curriculum but left to the teacher initiative in the 5th grade curriculum as of the 2017-2018 academic year and made it compulsory to be used in the "Information Technologies and Software" course. In the 2018-2019 academic year, it was gradually included in the 6th grade curriculum (MEB, 2018). Computer programming is a process that involves many skills (Tüzün, 2007). In this process, students can acquire problem solving, logical thinking, algorithm skills and even analytical thinking skills (Ersoy, Madran and Gülbahar, 2011). A programmer should first identify the problem, develop an algorithm for the solution, write the algorithm that they have developed into the code blocks and analyze possible errors. Therefore, the programming process can be perceived as a complex process. Because programming training requires a number of skills, it can be challenging for beginners (Gomes and Mendes, 2007).

Computer science course is constantly changing and developing despite not being one of the ongoing courses for years like science, social sciences or mathematics, which causes students to think that it is a rather complicated thing when they first meet programming (Nedzad and Yasmeen, 2001). Students who meet the computer programming environment consisting of abstract concepts for the first time have difficulty in programming training. Yet making these concepts concrete increases the motivation of students in this. In the study with two different groups within the scope of visualization studies in programming education carried out by Lahtinen and others (2007), it was seen that the group who had to prepare the programming assignments visually made their homework more regularly than the other group. In fact, programming education is difficult and boring and this leads to the development of prejudice against it in young individuals (Saygıner, 2017). Nowadays, on the other hand, with the help of block-based programming tools which are developed by using the developments in computer design, programming has turned into a fun educational environment that can be reached even by preschool children, young people and anyone who wants to learn programming (Genç and Karakuş, 2011).

Making programming easier and facilitating programming with block-based tools help to overcome the difficulties encountered in programming. However, in programming education, most processes and concepts may remain abstract for students. Nowadays, with the development of technology, robotic sets are used in programming education. Abstract concepts can be concretized by the programming of educational robot sets (Ersoy, Madran and Gülbahar, 2011). In this context, not only can educational robot applications and block programming environments contribute positively to the students' attitudes towards programming, and but also they can contribute to the development of critical thinking, problem solving, computational thinking and basic stem skills.

Developments in computer science have profound effects on economic and social life. Today, almost everyone, regardless of age, is expected to have some basic computational thinking skills in parallel with the new developments in technology (Wing, 2014). Having the



knowledge, skills and attitudes required to use the computer in solving daily life problems is called computational thinking (Özden, 2015). The purpose of using computational thinking in education is not only the increase of computer science qualifications of the students, but also making them gain the habit of using computational thinking skills in other courses (ISTE, 2015). These skills are defined as; creative thinking, algorithmic thinking, critical thinking, problem solving, cooperative learning and communication skills.

The basic stem skills of students can be improved by using educational robot applications and block programming environments. In the educational process from pre-school to higher education, science, technology, engineering and mathematics are brought together by an interdisciplinary approach to solve problems in everyday life, and this approach is defined as 'Stem approach' (Altunel, 2018). In this method, students' feelings of wonder are revived; they are expected to transform their research and inquiry-based learning into a product. Fidan and Yalçın (2012) state that educational robot applications improve students' mathematical thinking skills, collaborative study skills, creativity and problem solving skills, but also teach them scientific method, programming logic and engineering designing processes. It can be said that these skills are similar to computational thinking and Stem skills. Saritepeci and Durak (2017) emphasize that students should be prepared to solve the problems that we do not know now but we may encounter in the future using technologies. For many years, the basic education period has focused on children's reading, writing and mathematics learning. However, in recent years, Stem and science learning have begun to be emphasized (Gelman and Brenneman, 2004).

In recent years, robotic technologies have been used in education in order to develop basic knowledge and skills related to disciplines of Science, Technology, Engineering and Mathematics. Tools such as intelligent objects, virtual robot programming environments, self-made sets, which aim to give programming instruction to the students with a programmable physical robot, have become widespread (Numanoğlu and Keser, 2017). In the literature, it is possible to find many studies that educational robots and block programming contribute to the different cognitive characteristics of students (Saygıner, 2017; Demirkol, 2016; Yiğit, 2016; Erol, 2015; Kaucic and Asic, 2011; Kert and Uğraş, 2009). When the literature was examined, the effect of robot use in educational environments in recent years was also investigated thoroughly (Ching, Yang, Wang, Baek, Swanson and Chittoori, 2019; Simsek, 2018; Chen, at al., 2017; Leonard, at al., 2016; Noble Chaudhary, Agrawal, Sureka and Sureka, 2016; Yuen, at al., 2014). When examined variables are checked, computational thinking skills were described for the first time by Jeannette M. Wing in 2006.

Wing (2014) considers that computational thinking skills are essential for every individual towards the middle of the 21st century, just like reading, writing and basic math skills. This has led computational thinking to be an important research area in recent years. Today, individuals need to have the 21st century skills to invent and innovate, and the way to equip students with these skills is through Stem training (Roberts, 2012). Stem is a popular education concept. In recent years, countries have been building their curriculum on this concept. In this case it is important to look at what affects students' perceptions of Stem skill levels. Carbonaro, Rex and Chambers (2004) stated that Stem, coding, computing, and engineering skills are effective tools for project-based learning in which all are combined related to the use of robotic sets in education. However, there is not enough evidence in the literature as to whether all these variables are combined and especially in relation to project based environments to examine the effects of education given to educational robots on students' computational thinking and Stem skills.

In this study, these 2 skills were considered together and this space was tried to be filled. Thusly it is thought that the research will contribute to the literature. In conclusion, the aim of the study is to investigate the effect of project-based arduino educational robot applications on students' computational thinking skills and their perception of Stem skill levels.

## 1.2.Sub Problems

- (1) How are the students' Stem and Computational thinking skill levels in general?
- (2) What is the effect of project-based arduino educational robot applications on students' Stem and Computational thinking skill levels?

## 2.Method

### 2.1. Research Design

The quasi-experimental research design was used in the research with pretest- posttest control group. Since the students in the study could not be assigned in a neutral way, the study has a quasi-experimental design. General application of quasi-experimental designs in the field of educational technology is to compare the same groups of students or classes with different teaching strategies and to compare them over the dependent variables that are supposed to be investigated during the determined period (Dündar, Şahinkayaşı and Şahinkayaşı, 2017). “Project Based Arduino Educational Robot Applications” is the independent variable the effect of which on the experimental group is investigated. In the control group, students were taught based on the “Existing Information Technologies and Software (ITS)” course Teacher's Guide (MEB, 2018). The research design of the study is summarized in Table 1.

Table 1. Research Design of the Study

Group	Pretest	Teaching Method	Posttest
Experimental	Perceptions of Computational thinking Skill Levels +	<i>Project Based Arduino Educational Applications</i> + ITS course Teacher's Guide (2018)	PCTSL+PSSL
Control	Perceptions of Stem Skill Levels	ITS course Teacher's Guide (2018)	

### 2.2. Study group

The study group consisted of 6th grade students in two different secondary schools who are similar in terms of their socioeconomic status. Within the scope of the research, classes were assigned to experimental and control groups in an unbiased manner. There are 15 students in the control group and 18 students in the experimental group. The researcher worked as a teacher in the ITS courses in the 2018-2019 academic year in both groups. Demographic data of the students who constitute the experimental and control groups are given in Table 2.

**Table 2.** Demographic information of the groups

Groups	Gender		Total
	Female	Male	
Experimental	7	11	18
Control	8	7	15
Total	15	18	33

### 2.3. Data Collection Tools

In this study, "Computational thinking skill level" and "Perceptions of Stem skill levels" scales were used as data collection tools.

#### 2.3.1. Scale of Computational thinking Skill Levels:

In order to measure the students' computational thinking skills, "Computational thinking skill levels scale" adapted by Korkmaz, Çakır and Özden (2015) was used. The scale developed for measuring university students' computational thinking skills was also adapted by the same researchers to secondary school students later. During the adaptation study, the scale, which was previously developed in 5-point Likert type with 29 items, was reduced to 22 items. Cronbach alpha ( $\alpha$ ) values were calculated and confirmatory factor analyzes were performed for each factor in the scale of 5 factors. After confirmatory factor analysis, seven items with very low regression value were excluded from the original scale. After the confirmatory factor analysis on the 22-item scale, the observed values were found to be acceptable. The results of the reliability analysis of the scale in general and its factors are summarized in Table 3.

**Table 3.** Results of the reliability analysis of the scale and its factors

Factors	Number of items	Cronbach Alpha
Creativity	4	,640
Algorithmic Thinking	4	,762
Collaboration	4	,811
Critical thinking	4	,714
Problem solving	6	,867
Total	22	,809

#### 2.3.2. Basic Stem Skill Levels Scale

Basic STEM Skill Levels scale which was adapted by Korkmaz, Cakir, Ugur Erdoğan and Öner (In Press) was used to determine the students' Stem skill levels. They adapted the Basic STEM Skill Levels scale, which was previously designed to measure the STEM skill levels of university students, to the secondary school level. The scale is a 7-point Likert-type scale consisting of 23 items that can be grouped under 3 factors. The results of the reliability analysis of the scale in general and its factors are summarized in Table 4.

**Table 4.** Internal Consistency Coefficients of the BSSLS

Factor	Number of Items	Cronbach's Alpha
Science	11	0.899
Engineering and Technology	6	0.858
Mathematics	6	0.800
Total	23	0.940

## 2.4. Experimental Operations

The experimental and control groups in the selected schools were determined neutrally. The study was carried out in the Information Technology and Software (ITS) course which is compulsory in schools. The researcher is the teacher of the course at the same time. The course plans were prepared by the researcher before the application. Within this framework, the learning outcomes and class hours were taken into consideration in the field of learning. While the lesson plans were being created, the ITS course curriculum and MEB (2018) were used. The experimental implementation process is summarized in Table 5.

**Table 5.** Experimental Implementation Steps

Weeks	Groups			
	Experimental		Control	
Week 1	Implementation of Pretest			
Week 2	Problem Solving Concepts and Approaches			
Week 3				
Week 4				
Week 5	Using Block-Based Programming Tool			
Week 6	Block Based Robotic Programming Tool		Block-Based	Programming Tool
Week 7	Activities		Activities	
Week 8	Project Development with Block Based Robotic Programming Tool		Project Development with Block Based Programming Tool	
Week 9				
Week 10				
Week 11	Implementation of Posttest			

General aspects of the implementation are:

- The application was carried out in a total of 11 weeks. The instructional design prepared by the researcher was used in these weeks. In the first week, "Computational thinking skill levels" and "Stem skill level perceptions" scales were applied to the experimental and control groups before the application as pre-test.
- Unit of "Problem solving concepts and approaches" was presented with the same lesson plans to both groups in weeks 2, 3, 4. In these weeks the students were taught ;
  - Data types,
  - the concepts of fixed and variable,
  - using the concepts of fixed and variable in problem solving,
  - dividing a problem into sub-problems,
  - using basic functions in the problem solving process,
  - developing an algorithm for solving the problem,
  - testing the solution of an algorithm.
- In week 5, information about the Scratch program, which is a block-based programming tool, was given to both groups.
- In the 6th and 7th weeks, students from the experimental group were informed about the Arduino educational robot set program with the Scratch for Arduino block-based program.

Experimental group students were taught;

- The aim of the Arduino educational robotics set,
- sensors used in the set,
- arduino educational robotics set programming with Scratch.



- In these weeks, the control group students continued their programming studies with the Scratch block based program.
- In weeks 8, 9 and 10; firstly, a seminar was given to the students by the researcher. At the seminar; information about the project description, project steps, sample projects was given. The experimental group was divided into teams of 3 also considering the preferences of the students. The project topics determined by the researcher were randomly distributed to the students in the experimental group. The aim of the project topics' being determined by the researcher was to provide a limitation for the arduino educational robot sets, which is a very large area for the researcher to determine and to save time. A total of 6 project studies were conducted in the experimental group. The names of the projects prepared by the teams are given below:
  - Piano from Fruits
  - Smart home
  - Forest fire system
  - Automatic garden irrigation system
  - Seeing dustbin
  - Street lamp with level
- The control group students were also given a project preparation seminar by the researcher. Again, in light of student preferences, the control group was divided into teams of 3 and the project topics were randomly distributed to the students. In the control group, 5 project studies were carried out. The project topics prepared by the teams;
  - Maze game,
  - Ghost capturing game,
  - Animation of learning numbers
  - Pinball Game
  - Fish Catching Game
- Finally, "Computational thinking skill levels" and "Stem skill level perceptions" scales were applied to the groups as post-test.

The 3-week process in the experimental and control groups (weeks 8, 9 and 10) was planned by considering the process steps related to the project-based learning concept of Moursund (1999).

### 2.5. Data Analysis

Due to the number of different questions in the data collection tools, the raw scores obtained with each of the scales were converted into a system of 100 to be between 20 and 100. As the number of students in the experimental and control groups was less than 20, non-parametric tests were used for all sub-problems. In this context, the mean, standard deviation, minimum, and maximum, Mann-Whitney U tests were used for independent measurements.

### 3. Findings

The findings of the basic Stem skill levels of the students in the experimental and control groups before the application are summarized in Table 6.



**Table 6.** Stem Skill Levels of Students in General

	N	The lowest	The highest	$\bar{x}$	s.d.
Science	33	20,70	95,50	55,2788	18,84965
Engineering and Technology	33	4,70	99,40	51,5606	21,56674
Mathematics	33	23,70	94,70	58,6636	18,95268
STEM Total	33	17,30	92,60	55,1939	18,21668

When Table 6 is examined, it is observed that the self-perception scores of the students towards the basic Stem skills range from 17.30 to 92.60 and the average is 55.19. When the factor scores are examined, it is observed that the lowest average belongs to Engineering and Technology ( $\bar{x} = 51.56$ ), while the highest mean belongs to Mathematics factor ( $\bar{x} = 58.66$ ). Accordingly, it can be said that students' perception of basic Stem skills is low, the lowest factor points belong to Engineering and Technology, and the highest mean belongs to Mathematics factor. The findings of the students in the experimental and control groups regarding the level of computational thinking in general before the application are summarized in Table 7.

**Table 7.** Students' Computational thinking Skill Levels in General

	n	The lowest	The highest	$\bar{x}$	S.S.
Creative Thinking	33	45,00	100,00	80,1515	15,48613
Algorithmic Thinking	33	20,00	100,00	65,6061	17,03661
Collaboration	33	20,00	100,00	72,8788	21,75971
Critical Thinking	33	25,00	100,00	69,5455	18,25872
Problem Solving	33	40,00	100,00	74,6455	14,89184
Computational thinking total point	33	41,80	100,00	72,7485	10,68858

While Table 7 is examined, it is observed that the self-perception scores of the students in terms of computer-based thinking skill levels vary between 41,80 and 100 and the average is 72.74. When the factor scores are examined, the lowest mean belongs to algorithmic thinking ( $\bar{x} = 65.60$ ), while the highest average is related to creativity factor ( $\bar{x} = 80.15$ ). According to this, it is possible to say that the computer skills of the students are high, the lowest factor-related points belong to algorithmic thinking, and the highest mean is the creativity factor. The results of the Mann-Whitney U test performed to determine whether experimental and control groups are similar in terms of Stem skill levels prior to experimental procedures are summarized in Table 8.

**Table 8.** Findings about the Similarity of Stem Skills of Students Before Application

Factors	Group	n	Rank Average	Rank Total	U	p
Science	Experimental	18	16.11	290.0	119.0	0.562
	Control	15	18.07	271.0		
Engineering and technology	Experimental	18	14.20	261.0	90.0	0.103
	Control	15	20.00	300.0		
Mathematics	Experimental	18	15.42	277.5	106.5	0.301
	Control	15	18.90	283.5		
Stem Total	Experimental	18	15.58	280.5	109.5	0.356
	Control	15	18.70	280.5		

When the results of the Mann Whitney U Test, which was conducted in order to determine whether the secondary school students participating in the experimental process are similar in

terms of Stem skills before the experimental application, both for Stem skills total scores ( $U = 109.5$ ;  $p > 0.05$ ) Science ( $U = 119.0$ ;  $p > 0.05$ ), Engineering and Technology ( $U = 109.5$ ;  $p > 0.05$ ) and Mathematics ( $U = 106.5$ ;  $p > 0.05$ ), it was observed that there was no significant difference between skills. Accordingly, it can be said that the groups were similar in terms of Stem skills before the application. The results of the Mann-Whitney U test performed to determine whether the experimental and control groups were similar in terms of their skill level in terms of computational thinking before the experimental procedures are summarized in Table 9.

**Table 9.** Findings Related to Similarity of Students' Computational thinking Skills Before the Application

Factors	Group	n	Rank Average	Rank Total	U	p
Creative Thinking	Experimental	18	17.17	309.0	132.0	0.913
	Control	15	16.80	252.0		
Algorithmic Thinking	Experimental	18	15.47	278.50	107.5	0.316
	Control	15	18.83	282.50		
Collaboration	Experimental	18	17.53	315.50	125.5	0.730
	Control	15	16.37	245.50		
Critical Thinking	Experimental	18	14.97	269.50	98.5	0.185
	Control	15	19.43	291.50		
Problem Solving	Experimental	18	15.78	284.0	113.0	0.423
	Control	15	18.47	277.0		
Computational thinking total point						
Computational thinking total point	Experimental	18	15.78	284.0	113.0	0.426
	Control	15	18.47	277.0		

When the results of the Mann Whitney U Test are examined which was performed to determine whether secondary school students participating in the experimental process were similar in terms of their computational thinking skills before the experimental application, it was observed that there was no significant difference between the total scores of both the experimental and control groups regarding the factors of computational thinking skills ( $U = 113.0$ ;  $p > 0.05$ ) as well as Creative thinking ( $U = 132.0$ ;  $p > 0.05$ ), Algorithmic thinking ( $U = 107.5$ ;  $p > 0.05$ ), Collaboration ( $U = 107.5$ ;  $p > 0.05$ ), Critical thinking ( $U = 98.5$ ;  $p > 0.05$ ) and Problem solving ( $U = 113.0$ ;  $p > 0.05$ ). According to this, it can be said that groups were similar in terms of computational thinking skills before the application. The results of the Mann-Whitney U test are used to differentiate between the experimental and control groups in order to determine the effects of the activities performed in the experimental and control groups reflecting the posttest scores on the students' Stem skills are summarized in Table 10.

**Table 10.** The Contribution of Experimental Practice to Students' Stem Skills

Factors	Group	n	Rank Average	Rank Total	U	p
Science	Experimental	18	19.00	276.0	105.0	0.277
	Control	15	15.33	285.0		
Engineering and technology	Experimental	18	14.72	265.0	94.0	0.138
	Control	15	19.73	296.0		
Mathematics	Experimental	18	15.47	278.50	107.5	0.319
	Control	15	18.83	282.50		
Stem Total	Experimental	18	14.97	269.50	98.5	0.187
	Control	15	19.43	291.50		

When the results of the Mann Whitney U Test are examined in table 10, to see whether the block-based robotics programming tool based on the posttest scores were significantly higher

than the activities based on block based programming tools for middle school students, it was determined that there was no significant difference between the scores and scores of the factors. Accordingly, it can be said that activities based on block-based robotics programming tools do not significantly contribute more to Stem skills of secondary school students compared to block based programming tool activities. The results of the Mann-Whitney U test for the differentiation between the experimental and control groups in order to determine the effect of the activities performed in the experimental and control groups according to the posttest scores on the computational thinking skills of the students are summarized in Table 11.

**Table 11.** Contribution of Experimental Application to Students' Computational thinking Skills

Factors	Group	n	Rank Average	Rank Total	U	p
Creative Thinking	Experimental	18	16.92	304.50	133.50	0.956
	Control	15	17.10	256.50		
Algorithmic Thinking	Experimental	18	17.14	308.50	132.50	0.927
	Control	15	16.83	252.50		
Collaboration	Experimental	18	16.75	301.50	130.50	0.869
	Control	15	17.30	259.50		
Critical Thinking	Experimental	18	17.75	319.50	121.50	0.620
	Control	15	16.10	241.50		
Problem Solving	Experimental	18	20.14	362.50	78.50	0.039*
	Control	15	13.23	198.50		
Computational thinking Total Point	Experimental	18	18.86	339.50	101.50	0.025*
	Control	15	14.77	221.50		

When the results of the Mann Whitney U Test, which was conducted to define whether activities based on block-based robotics programming tools significantly contributed to the computer-based thinking skills of the students compared to the activities based on the block-based programming tools, are examined (please see table 11), there was a significant difference between the total scores of computational thinking skills of the experimental and control groups ( $U = 101.50$ ,  $p > 0.05$ ). When it comes to the points related to factors, it was determined that there was a significant difference only between problem solving skills ( $U = 78.50$ ,  $p > 0.05$ ). Accordingly, it can be said that activities based on block-based robotics programming tools contribute significantly to the total score and problem-solving factor compared to the computational thinking skills and activities based on block-based programming tool.

#### 4. Discussion and Conclusion

When the students' perception levels about their basic Stem skills are focused upon, they appear slightly low, the points for the lowest factor being Engineering and Technology, and the highest is the Math factor. According to this, it can be said that the students did not have much experience in engineering and technology studies, and that the mathematics courses given in the school provided students with high mathematics perceptions. It was determined that there was no significant difference between the experimental and control groups compared to the Tests of Basic Stem skills levels before the experimental invention, in other words, the groups were similar before the application. When the differentiation between the experimental and control groups is examined in order to determine the effects of the activities performed in the experimental and control groups on the students' Stem skills in light of the posttest scores, it was determined that there was no significant difference between

the experimental and control groups in terms of both total scores and the scores related to the factors. In this context, it can be said that the activities based on block-based robotics programming tool conducted within the scope of ITS course did not contribute significantly to Stem skills of secondary school students compared to the activities based on block based programming tool. Leonard and others (2016) in their study on secondary school students observed that educational robotic sets and game designs increased the students' thinking skills but did not change their attitude towards Stem. Again according to Leonard and others (2016) this is due to the limitations of the study and the loss of the subject. 124 people participated in the study, but 76 people took the questionnaire. There were also cultural differences (American, Indian, and natives) among the participants. Unfried, Faber, Stanhope, and Wiebe (2015) suggest in their study the evaluation of stem attitudes calls for a long process as it requires time for students to adopt Stem.

Benitti (2012) stated that there should be activities that integrate interdisciplinary curriculum and robotics into other courses (science and engineering), and educational robot applications aiming to improve the knowledge of Stem will help students who has interest in robotics as well as Stem. In this context, it can be said that both educational robot activities and block programming activities contributed significantly to the students' Stem skills and these contributions were similar for both applications. In this context, it can be said that both applications can be used to improve students' Stem skills. It is possible to say that the students have high levels of computational thinking skills, the lowest factor points belong to algorithmic thinking, and the highest average is related to the creativity factor. When the computational thinking skills tests were compared before the experimental application, it can be said that there was no significant difference between the experimental and control groups, in other words the groups were equal before the application. When the differentiation in groups about project-based educational arduino robot activities performed in the experimental and control groups was examined according to the posttest scores with a view to determining the effect of the activities on the computational thinking skills of the students, it was found out that there was a significant difference between the total scores and the points related to the problem solving factor. In this context, it can be said that activities based on block-based robotics programming tool conducted within the scope of ITS course contributed more significantly to the computational thinking skills of secondary school students compared to their activities based on block based programming tool.

Atmatzidou and Demetriadis (2016) stated in their study of the effect of educational robotic sets on students' computational thinking skills, that the students' scores increase significantly towards the end of the activity as, it requires time to fully develop computer-thinking skills, and the use of different methods (written and oral) may have an impact on student performance when evaluating computational thinking skills. Chen and others (2017) say that many of the existing computer-based assessments focus more on studying student products after learning a specific platform. They suggest that this limitation of evaluation method leads to the interpretation of computational thinking as a basic skill that can be transferred between platforms. They state that separate evaluation tools should be prepared for these platforms. They also argue that there are 4 basic levels (Block-based environments, a creative hybrid environment for robotics programming, VEX robotic design systems, and Dash) that should be evaluated with appropriate tools. On the other hand, Chen and others (2017) state in their research that the sub-skills of algorithmic thinking did not improve because in the visual programming environment used, the syntax was not as clear as in other languages (Java, C, etc.) that were syntactically more stringent. Henceforth in this context, activities based on educational robots can be preferred in order to increase students' thinking skills instead of

block-based programming activities.

## 5. Suggestions

- The use of both block-based robotic activities and block-based programming activities can be recommended for students to improve their Stem skills
- It may be advisable to focus on the use of block-based robotic activities in order to develop students' computational thinking skills.

## 6. Acknowledgement

- This article was produced from the master thesis same titled, written by Kübra Karaahmetoğlu, under the supervision of Özgen Korkmaz.
- This research was supported by Scientific Research Project Coordinator of the University of Amasya, under project number SEB-BAP 18-0166.

## References

- Akgündüz, D., Aydeniz, M., Çakmakçı, G., Çavaş, B., Çorlu, M., Öner, T., & Özdemir, S. (2015). A report on STEM Education in Turkey: A provisional agenda or a necessity? İstanbul: İstanbul Aydın University STEM Center.
- Altunel, M. (2019). STEM Education and Turkey: Opportunities and Risks, Available at: [https://www.setav.org/perspektif-stem-egitimi-ve-turkiye-firsatlar-ve-riskler/, Retrieved: 10.03.2019.]
- 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.
- Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, 58(3), 978-988.
- Carbonaro, M., Rex, M., & Chambers, J. (2004). Using LEGO robotics in a project-based learning environment. *The Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, 6(1).
- Chaudhary, V., Agrawal, V., Sureka, P., & Sureka, A. (2016, December). An experience report on teaching programming and computational thinking to elementary level children using lego robotics education kit. In *2016 IEEE Eighth International Conference on Technology for Education (T4E)* (pp. 38-41). IEEE.
- Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., & Eltoukhy, M. (2017). Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Computers & Education*, 109, 162-175.
- Ching, Y. H., Yang, D., Wang, S., Baek, Y., Swanson, S., & Chittoori, B. (2019). Elementary School Student Development of STEM Attitudes and Perceived Learning in a STEM Integrated Robotics Curriculum. *TechTrends*, 1-12.
- Demirkol, Z. (2016). Çocuklar için kodlama [Coding for kids], İstanbul: Pusula Pub.
- Erol, O. (2015). The Effect of Scratch Programming Programming on Motivation and Achievement of Information Technology Teachers. (PhD. Thesis), Anadolu University, Institute of Educational Sciences.
- Ersoy, H., Madran, R. O., & Gülbahar, Y. (2011). Proposing a model for teaching programming languages: robot programming. *Academic Informatics Conference*.



- Gelman, R., & Brenneman, K. (2004). Science learning pathways for young children. *Early Childhood Research Quarterly* (Special Issue on Early Learning in Math and Science), 19(1), 150–158.
- Genç, Z. & Karakuş, S. (2011). Learning through design: using scratch in instructional computer games design. 5th International Computer & Instructional Technologies Symposium. Elazığ
- Gomes, A., & Mendes, A. J. (2007, September). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE* (Vol. 2007).
- Fidan, U., & Yalçın, Y. (2012). Lego Nxt Training Kit. *Afyon Kocatepe University Journal of Sciences*, 12(1), 1-8.
- Karabak, D. ve Güneş, A. (2013). Curriculum Proposal for First Class Secondary School Students in The Field of Software Development. *Journal of Research in Education and Teaching*, 2(3), 163-169.
- Kaucic, B., & Asic, T. (2011). Improving introductory programming with Scratch?. In *MIPRO, 2011 Proceedings of the 34th International Convention* (pp. 1095-1100).
- Kert, S. B., & Uğraş, T. (2009). Simplicity and fun in programming: Scratch examples. In *The First International Congress of Educational Research*, Çanakkale, Turkey.
- Korkmaz, Ö., Çakır, R., ve Özden, M. Y. (2015). Computational thinking levels scale (ctls) adaptation for secondary school level. *Gazi Journal of Education Sciences*, 1(2), 67-86.
- Korkmaz, Ö., Çakır, R., Uğur Erdoğan, F. & Öner, F. (In press). Secondary School Students' Basic STEM Skill Levels according to their Self-Perceptions: A Scale Adaptation Study. *International Journal of Science and Mathematics Education*
- Lahtinen, E., Ahoniemi, T. ve Salo, A. (2007). Effectiveness of integrating program visualizations to a programming course. *Proceedings of the 7th Baltic Sea Conference on Computing Education Research*, 195- 198. Koli, Finland.
- Leonard, J., Buss, A., Gamboa, R. ve ark. *J Sci Educ Technol* (2016) 25: 860. <https://doi.org/10.1007/s10956-016-9628-2>
- International Society for Technology in Education (ISTE). (2015). CT Leadership toolkit. Available at: [Çevrim-ici: <http://www.iste.org/docs/ct-documents/ctleadership-toolkit.pdf?sfvrsn=4>, Erişim Tarihi: 08.04.2019.]
- Ministry of Education (MEB). (2018). Information technology and software course curriculum (5th and 6th grade secondary schools). Available at: [Çevrim-ici: <http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bilism%20Teknolojileri%20ve%20Yazilim%205-6.%20Siniflar.pdf>, Retrieved: 08.05.2019.]
- Ministry of Education (MEB). (2018). The 6th Grade ITS lesson Teacher's Guide. <http://www.eba.gov.tr/ekitap?icerik-id=6696>
- Nedžad M, & Yasmeen H (2001) Challenges in teaching Java technology. *Challenges Informing Clients A Transdiscipl Approach* 365–371
- Noble, J. (2013). Building a LEGO-based Robotics Platform for a 3 rd Grade Classroom, Doctoral dissertation, Tufts University.
- Numanoğlu, M., & Keser, H. (2017) Robot Usage in Programming Teaching - Mbot Example *Bartın University Journal of Faculty of Education*, 6(2), 497-515.
- Özden, M. Y. (2015). Computational thinking. Available at: [Çevrim-ici: <http://myozden.blogspot.com.tr/2015/06/computational-thinking-bilgisayarca.html>, Erişim Tarihi: 06.03.2019.]
- Roberts, A. (2012). A Justification for STEM education. technology and engineering teacher. Available at: [Çevrim-ici:

- <http://www.iteaconnect.org/mbrsonly/Library/TTT/TTTe/04-12roberts.pdf> /, Erişim Tarihi: 13.02.2019.]
- Sarıtepeci, M., & Durak, H. (2017). Analyzing the effect of block and robotic coding activities on computational thinking in programming education. *Educational research and practice*, 490-501.
- Saygıner, Ş. (2017) Effects Of Block-based Visual And Text-based Programming Instruction On Achievement, Logical Thinking And Motivation. (Master Thesis) Hacettepe University, Institute of Educational Science.
- Somyürek, S. (2014). Gaining the Attention of Generation Z in Learning Process: Augmented Reality. *Educational technology, Theory and Practice*. 4(1), 63-80.
- Dündar, F., Şahinkaya, Y., & Şahinkaya, H. (2017). Perceptions of Lower Level EFL Students on Corpus-based Grammar Learning. *Electronic Turkish Studies*, 12(34).
- Şimşek, E. (2018). The effect of robotics and scratch applications on students' computational thinking skills and academic achievement in programming teaching. (Master Thesis). 19 Mayıs University, Institute of Educational Science.
- Tüzün, H. (2007). Programming 2.0: the use of innovative Internet technologies in programming education. *Academic Informatics Conference Kütahya: Dumlupınar University*.
- Unfried, A., Faber, M., Stanhope, D. S., & Wiebe, E. (2015). The development and validation of a measure of student attitudes toward science, technology, engineering, and math (S-STEM). *Journal of Psychoeducational Assessment*, 33(7), 622-639.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33-35.
- Wing, J. M. (2014). Computational thinking benefits society. 40th Anniversary Blog of Social Issues in Computing, 2014.
- Yenilmez, K. & Balbağ, M. Z. (2016). The Stem Attitudes of Prospective Science and Middle School Mathematics Teachers. *Journal of Research in Education and Teaching*.5(4), 301- 307.
- Yiğit, M. F. (2016). *Visual programming environment and the impact of teaching on students' attitudes towards computer programming and programming*. (Master Thesis). 19 Mayıs University Institute of Educational Science
- Yuen, T., Boecking, M., Stone, J., Tiger, E. P., Gomez, A., Guillen, A., & Arreguin, A. (2014). Group tasks, activities, dynamics, and interactions in collaborative robotics projects with elementary and middle school children. *Journal of STEM Education*, 15(1).