Baltic J. Modern Computing, Vol. 4 (2016), No. 4, 647-660

The Effect of Lego Mindstorms Ev3 Based Design Activities on Students' Attitudes towards Learning Computer Programming, Self-efficacy Beliefs and Levels of Academic Achievement

Özgen KORKMAZ

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

ozgenkorkmaz@gmail.com

Abstract. The aim of the study was to present the effect of Lego Mindstorms Ev3 based design activities on students' attitudes towards learning computer programming, self-efficacy beliefs and levels of academic achievement. The research is a pretest-posttest with control group semi-experimental study. The study group of the research consisted of 53 students attending the faculty of engineering. The experimental group was administered a teaching method with Lego Mindstorms Ev3 based design activities. On the other hand, in the control group C++ topics were directly taught via editor. Research data were collected via the implementing academic achievement test (Kr-20= 0.71), attitude towards learning programming scale (Cronbach's Alpha =0.84) and computer programming self-efficacy scale (Cronbach's Alpha= 0.966). Obtained findings were: The education using Lego Mindstorms Ev3 design activities contributed more meaningfully to the academic success of the students in terms of C++ programming language compared to the traditional method.

Key Words: Lego Mindstorms Ev3, programming, attitude, academic achievement, and self-efficacy perception

1. Introduction

No matter what the system of learning, the word "game" usually attracts the attention of students. The reason is that voluntariness is the basis for playing games (Huizinga, 2013). In addition, the fact that games make students feel included and games are activities that give them experiences of motivation, problem solving, embracing, flow, creativeness and action (Prensky, 2003) always makes the games attractive. Accordingly, many educational play instruments, toys and digital games have been produced and continue to be produced. An internationally renowned instrument of play, Lego, was first produced by Danish carpenter Ole Kirk Christiansen making wooden toys and has evolved from the idea of toys until today when it has become prominent by taking on an amusing and educational dimension (Güntürkün, 2009). In time, modular bricks made from wood were transformed into plastic with kits from which controllable and programmable robots could be produced.

Lego robotic systems comprise Lego bricks, a programming language, a microprocessor, wheels and friction gears. Moreover, Logo has developed a programming language that can work in cooperation with Lego bricks. The visual programming language based on the Lego system has been transformed into RoboLab over the years. Having been seen as a boring and repetitive activity for many people because of lack of motivation, inappropriate teaching methods and low interaction, programming has been turned into a coherent and amusing subject since it uses flow diagrams instead of written text. Thus sensors, motors, Lego bricks and programmable bricks became the basis of Lego Mindstorms (Güntürkün, 2009). Lego Mindstorms is a robotic set created by Lego for children aged 10 and above. It is aimed at getting students to acquire programming, basic design and robotic principles. Robots attached to the bricks in the set are programmed with Robotic Invention System-RIS, which is a visual software language, or RoboLab software, then operated by loading to an RCX known brick using USB cable or infrared (Varnado, 2005). EV3 brick has taken the place of RCX since 2013. This brick has four motor ports, one micro card slot, USB slot and Linux operating system. Furthermore, there are more buttons in the system that has an LSD screen in larger size and higher resolution. There are more programming tools in the programming interfaces of EV3.

According to Çayır (2010), Lego has a lot of advantages such as presenting students real life experiences, providing opportunities to work in groups or individually, making students participate in the process of learning actively, making them use their interdisciplinary knowledge, and contributing to presenting alternative ways in the process of problem solving. It is simple to find evidence proving that robotic applications enable students to learn concepts in science, technology and engineering classes easier in the literature (Moore, 1999; Papert, 1980). Additionally, it is possible to see much evidence proving that robotic applications contribute to the problem solving abilities of students, multidimensional thinking abilities and logical thinking abilities and increase motivation (Varney et al., 2012 ; Zaharija et al., 2013).

The important thing in computer programming is not what orders to input, but the solution of a problem by making an algorithm with the orders. Accordingly, the process of programming is a problem solving process. Therefore, it is closely related to high level thinking abilities such as problem solving, logical and mathematical thinking, critical thinking and creative thinking (Korkmaz and Altun, 2014; Fang, 2012; Korkmaz, 2012; Lau and Yuen, 2009; Wang et al., 2012; Czerkawski and Lyman, 2015). It is possible to encounter many studies in the literature concerning difficulties in the teaching and learning of computer programming (Korkmaz and Altun, 2014; Gomes and Mendes 2007; Tan et al., 2009; Jenkins, 2002; Katai et al., 2008; Korkmaz, 2012; Korkmaz, 2013; Milne and Rowe 2002). The most prominent among these difficulties is that individuals do not have these high level thinking abilities. It is possible to benefit from Lego robot sets in the process of programming education as in many other areas because it provides a more concrete programming environment and gives support to the improvement of high level thinking abilities. It is possible to see in the literature that Lego robot sets are utilized in teaching subjects such as science, mathematics and basic engineering concepts. However, we have not encountered enough evidence in the literature related to whether it contributes to teaching direct programming education at a specialized bachelor degree level. In this context, the aim of the study has been identified as presenting the effect of Lego Mindstorms Ev3 based design activities on students’ attitudes towards learning computer programming, self-efficacy beliefs and levels of academic achievement.

There is sufficient evidence suggesting that attitude is effective in many cognitive and affective characteristics about learning (Hwang et al., 2012; Lai et al., 2012). It has been stated that especially negative perception, motivation and attitude are more important factors than the other factors affecting academic success (Anastasiadou and Karakos, 2011; Erdogan et al., 2008). Self-efficacy is defined as the belief in achieving a job. It has been stated that whether this belief is connected to the behavior related to the job or not affects motivation and academic success as a consequence (Kotaman, 2008). Therefore, self-efficacy beliefs and attitudes towards learning programming as well as academic success have been discussed in the study.

1.1. Problem Statement:

Do Lego Mindstorms Ev3 based design activities have any effect on students' attitudes towards learning computer programming, self-efficacy beliefs and levels of academic achievement?

1.2. Sub Problems:

a. On a general scale, what are the students' academic achievements, attitudes towards learning computer programming and programming self-efficacy beliefs?

b. Does gender change students' academic achievements in programming, attitudes towards learning programming and self-efficacy beliefs?

c. Do traditional methods and Lego Mindstorms Ev3 based design activities change students' academic achievements in programming, attitudes towards learning programming and self-efficacy beliefs?

2. Methods

2.1. Research Design

The research is a pretest-posttest with control group semi-experimental study. The graphic image of the test model employed in the research is as below:

Table1. Experimental Design

Groups	Pretest	Experimental Manipulation	Posttest
Experimental Group	• Academic Achievement Test • Attitude towards learning programming scale • Computer programming self-efficacy scale	Lego Mindstorms Ev3 based design activities	• Academic Achievement Test • Attitude towards learning programming scale • Computer programming self-efficacy scale
Control Group		Traditional Method	

2.2. Study Group

The study group of this research is composed of 53 students. Of the study group taking the 3-credit Computer Programming course in the Faculty of Engineering Department of Computer Engineering and Electric –Electronics Engineering Department 1st grade, 12 students are females and 41 are males in both sections. While creating Experimental Groups, no intervention was made to the groups and natural classroom structures in schools were preserved. Random selection method was administered in designating the groups as test or control groups. When unevaluated participants were excluded, distribution of study groups with respect to gender became as summarized in Table 2.

Table2. Distribution of Study Groups with respect to Gender

Groups	Male	Female	Total
Experimental Group	21	6	27
Control Group	20	6	26
Total	41	12	53

2.3. Experimental Procedures

Experimental Group: Teaching based on Lego Mindstorms Ev3 design activities was applied to the Experimental Group. In this context, Lego Mindstorms Ev3 was introduced to students for two weeks and design and programming elements in this environment were explained by example. The two projects in the guidelines received with the Lego Mindstorms Ev3 set were programmed by designing with students. The students were asked to make a unique design and program it by using Lego Mindstorms Ev3 until the last week of the application. Since each student could not be provided with a Lego Mindstorms Ev3 set to make these designs, group work was done. In this context, as there were eight robots available, the students were grouped on a voluntary basis. Thus eight groups were made in total; four groups each consisting of four or three persons.

Then, the subjects of variables in C++, basic structure of C++, program controls, loops, conditions, functions and basic built-in functions were discussed for six weeks and examples of each subject were shown on both Lego Mindstorms Ev3 and C++ editor. In addition, the students were given feedback and corrections about Lego Mindstorms Ev3 projects each week. At the end of six weeks, a project competition was organized regarding the design responsibilities given to students at the beginning of the application and the projects were evaluated both by the students and the researcher in terms of both product and programming processes. The students ranked among the top three were rewarded. Example views of the project work of the students have been presented in Figure 1.

Control Group: In the Control group, identical pretest and posttest practices were administered in the first and last weeks. In the remaining eight-week period, the very same C++ subjects were lectured. Upon giving examples of each subject, C++ editor was used to solve the sample cases. Also, students were given homework assignments and before each lesson assignments were checked in class. Unsolved questions were explained in the classroom.

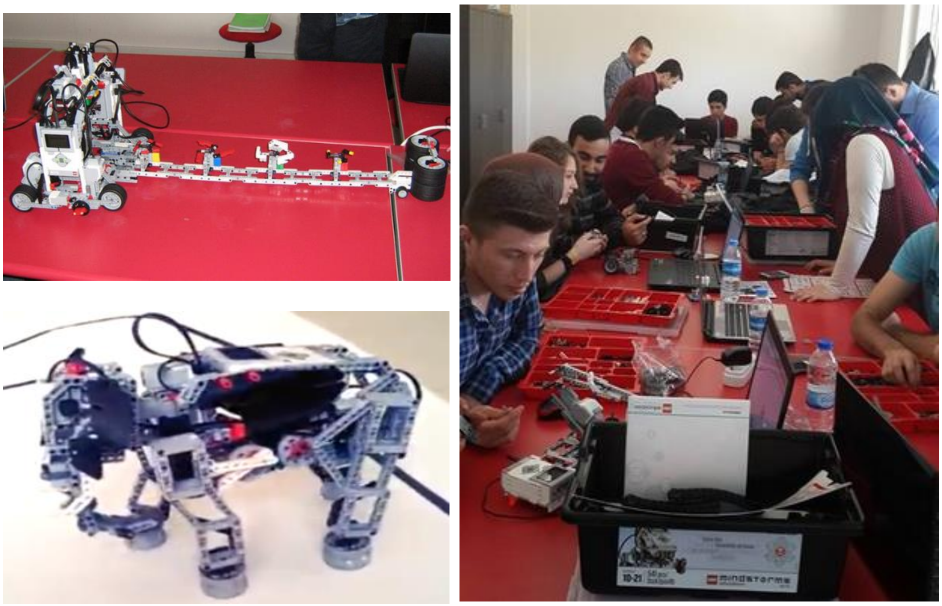

Figure 1. Lego Mindstorms Ev3 Project Works

2.4. Data Collection Tools

Academic Achievement Test: Academic achievement test was performed within the scope of present research. 30 items with 5-choice questions were prepared containing topics such as variables in C++, basic structure of C++, program controls, loops, conditions, functions and basic available functions. As a result of pilot study conducted on 148 students who had previously taken C++ course five items of which had a distinctiveness level below 0.3 and whose exclusion would not disrupt content validity were removed from the scale. In its final version the scale contained 25 items and distinctiveness coefficients of the items varied between 0.296 and 0.740 and its difficulty index was measured as 0.51. Kr-20 internal consistency coefficient was measured as 0.71.

Attitude towards learning programming scale: Research data were compiled via "Attitude towards Computer Programming Scale" developed by Korkmaz and Altun (2014a). Attitude towards Computer Programming Scale was developed by Korkmaz and Altun (2014a) to detect students' attitudes in terms of validity and reliability. The scale consisted of total 20 items and three factors. Each single item was scaled as; "Never (1), Rarely (2), Sometimes (3), Generally (4), Always (5)". Construct validity and internal consistency information of the scale are as summarized in Table 3.

Table 3. Construct Validity and Internal Consistency

Factors	Items	KMO	Bartlett	Eigenvalue	Explained Variance	Cron. Alpha
F1 (Motivation)	9	0,876	x^2=2867,942; sd=190; p<0,001	5,70	17,55	0,82
F2 (Negative Attitude)	6			2,10	16,01	0,77
F3 (Necessity)	5			1,61	13,69	0,75
Total	**20**			-	**47,34**	**0,87**

Table 4 shows that KMO value is 0.876 and Bartlett value is below 0.05. Internal consistency coefficients are, for all the factors and total score, above 0.70. Within this framework it is safe to claim that the scale is a valid and reliable tool in measuring students' attitudes towards learning computer programming.

Computer programming self-efficacy scale: In order to designate students' self-efficacy beliefs towards programming, Computer Programming Self-efficacy Scale developed by Ramalingam and Wiedenbeck (1998) and adapted into Turkish by Korkmaz and Altun (2014b) was utilized. In line with adaptation analyses made by Korkmaz and Altun (2014b) this scale originally composed of 32 items was decreased to 28 items and one factor. Factor loads of scale items varied between 0.618-0.807. Item factor correlations are between 0.621 and 0.804, corrected correlations are between 0.588 and 0.779. Internal consistency coefficient is (Cronbach's Alpha) 0.966. Accordingly the scale is, in Turkish culture as well, a valid and reliable tool to measure students' self-efficacy beliefs towards computer programming.

2.5. Data Analysis

Data collection tools employed in the research failed to be standard due to the scores corresponding to the responses given to 5-, and 7-Likert type scales and differentiations in the number of items in subfactors. Therefore, the best method seemed to convert the obtained raw scores into standard scores the lowest of which would be 20, and the highest 100. Accordingly, prospective teachers' attitudes towards the teaching profession were examined by using frequency, percentage, arithmetic means, standard deviation, and t and Anova analyses. On the other hand students having received 46 or lower scores were categorized in the lower level, between 47 and 72 as average level and those having received 72 and higher scores as in higher-order category.

3. Findings

3.1. Students' Attitudes towards Learning Computer Programming, Self-efficacy Beliefs and Levels of Academic Achievement

Students' Attitudes towards Learning Computer Programming, Self-efficacy Beliefs and Levels of Academic Achievement have been generally examined according to post test points. In this context, the findings related to academic success have been summarized in Table 4.

Table 4. Academic Success of Students

Acad. Achiev.	N	Min	Max	Mean	Std. Dev.	Skew.	Kur.
Cont. Group	27	8,00	80,00	45,2	16,0	-0,302	0,542
Exp. Group	26	20,00	84,00	66,8	16,9	-1,376	1,118
Total	53	8,00	84,00	55,4	19,4	-0,494	-0,541

When the post test points of academic success in Table 4 have been examined, the lowest point students have received is 8 and the highest point is 84 out of 100. The general academic average of students is 55.4. Accordingly, it can be said that academic success of the students related to C++ programming is generally at the medium level and is not high enough. This situation might have been caused by the fact that the students haven't been supported enough in high level thinking abilities, they consider C++ programming irrelevant for their occupation, their attitudes towards learning is low and self-efficacy beliefs are weak. On the other hand, when Skewness and Kurtosis values have been examined, it has been seen that these values range between -0.302 and -1.376. According to the analysis, Shapiro-Wilk is 53=0.950 and p<0.05. According to Shapiro-Wilk analysis, it has been seen that academic success grades in total are not in a normal distribution and it can be said that all Skewness and Kurtosis values range between -1.5 and +1.5 and according to Tabachnick and Fidell (2013), this value range is proof of normal distribution. Moreover, George and Mallery (2010) have considered that it is enough for these values to be between -2 and +2. The findings concerning the attitudes towards students' learning with program and the levels of self-efficacy beliefs have been summarized in Table 5.

Table 5. Students' Attitudes Towards Learning Programming and Self-efficacy Beliefs

Variables	N	Min	Max	Mean	Std. Dev.	Skew.	Kur.	Levels(f/%)		
								Low	Med.	High
Self-Efficacy	53	23	98	65,9	17,3	,-25	-,45	9 %17,0	20 %37,7	24 %45,3
Motivation		38	100	75,8	15,8	-,43	-,53	2 %3,8	19 %35,8	32 %60,4
Negative Attitude		20	100	78,8	20,5	-,67	-,47	3 %5,7	15 %28,3	35 %66,0
Necessity		20	76	32,2	16,8	1,29	,56	44 %83,8	7 %13,2	2 %3,8

As seen in Table 5, it has been observed that the points of self-efficacy beliefs of the students range between 23 and 98 and the average is 65.9. On the other hand, it has been seen that 17.0% of the students have low self-efficacy and 37.7% of the students have medium self-efficacy; however, 45.3% of the students have high self-efficacy. Accordingly, it can be said that a great part of the students feel themselves competent at C++ programming.

When the attitudes towards learning C++ programming are examined, the lowest point in the willingness factor is 38 and the highest point is 100. The average is 75.8. On the other hand, 60% of the students are willing on a high level about learning C++ programming and 35.8% of them are willing at a medium level. Only 3.8% of the students are unwilling. The lowest point in the negative perception factor is 20 and the highest 100. The average is 78.8. On the other side, 66.4% of the students have negative perception at a high level about learning C++ programming and 28.3% of them have medium-level negative perception. The lowest point in the necessity factor is 20 and the highest 76. The average is 32.2. On the other hand, 83.8% of the students find the subject of learning C++ programming necessary at low level and 13.2 % of them find it necessary at medium level. Accordingly, it can be said that an important part of the students are willing to learn C++; however, the other important part have negative thoughts about learning C++. In addition, an important part of the students can be said to find learning C++ unnecessary.

Moreover, normal distribution has been observed in the points of both self-efficacy and attitude factors. The findings related to the students' academic success, attitudes towards learning programming and the levels of self-efficacy beliefs in terms of gender have been summarized in Table 6.

Table 6. Students' Academic Success, Attitudes towards Learning Programming and The Levels of Self-efficacy Beliefs in Terms of Gender

Variables	N	Mann-Whitney U	Standart Error	p
Akademic Ach.	53	242,0	46,905	0,932
Self-Efficacy		302,0	47,009	0,234
Motivation		317,5	46,976	0,128
Negative Attitude		293,5	46,451	0,307
Necessity		211,0	43,414	0,420

When Table 6 was analyzed, it has been seen that genders of the students are not effective in the students' academic success, attitudes towards learning programming and the levels of self-efficacy beliefs. In other words, success, self-efficacy and attitudes of both male and female students are similar.

3.2. Findings as regards the Equivalence of the Groups Before the Application

Whether the groups are equivalent in terms of academic success, attitudes towards learning programming and also self-efficacy before the application or not has been searched in Table 7.

Table 7. Findings as regards the Equivalence of Groups

Variables		N	$\bar{X}$	S.S	t	Df	p
F1 (Notivation)	Control Gr	27	73,1	14,1	0,552	51	0,583
	Experimental Gr	26	70,9	14,9			
F2(Negative Attitude)	Control Gr	27	72,4	16,2	-0,692		0,492
	Experimental Gr	26	75,5	15,9			
F3(Necessity)	Control Gr	27	29,2	13,4	-1,29		0,201
	Experimental Gr	26	35,4	20,8			
Academic achievements	Control Gr	27	30,5	10,9	0,991		0,326
	Experimental Gr	26	27,1	14,2			
Self-efficacy	Control Gr	27	42,1	18,3	-0,271		0,787
	Experimental Gr	26	43,4	14,6			

When Table 7 was examined, it has been observed that there are small differences in the academic success, attitudes (willingness, negative perception, necessity) towards learning programming and the self-efficacy beliefs of test and control groups, yet these differences are not meaningful. Accordingly, it can be said that the groups are equivalent in terms of academic success, attitude and self-efficacy before the application. Although the differentiations between the groups are not meaningful, post test-pre test difference points have been used in the analysis below with the purpose of bringing under control the effect of small differences.

3.3. The Effectiveness of LEGO Mindstorms Ev3 Design Activities

The findings as regards whether there have been differences in terms of academic success, attitude and self-efficacy between the traditional method based on C++ editor and the Lego Mindstorms Ev3 design activities have been summarized in Table 8.

Table 8. Effectiveness of Lego Mindstorms Ev3 Design Applications

Variables		N	$\bar{X}$	S.S	t	df	p
F1 (Notivation)	Control Gr	27	-1,7	21,6	-1,296	51	0,201
	Experimental Gr	26	7,3	28,8			
F2(Negative Attitude)	Control Gr	27	6,4	16,7	0,578		0,566
	Experimental Gr	26	3,3	21,6			
F3(Necessity)	Control Gr	27	3,4	17,1	1,189		0,240
	Experimental Gr	26	-4,0	27,3			
Academic achievements	Control Gr	27	14,7	16,6	-4,235		0,000
	Experimental Gr	26	38,9	24,5			
Self-efficacy	Control Gr	27	22,5	18,6	-0,212		0,833
	Experimental Gr	26	23,8	21,7			

When Table 8 was examined, it has been seen that Lego Mindstorms Ev3 design activities have not caused any meaningful differentiation in the students' attitudes towards learning programming, willingness ($t_{(2-51)}$=-0.201; p>0.05), negative perception ($t_{(2-51)}$=-0.566; p>0.05) and necessity ($t_{(2-51)}$=-1.189; p>0.05) compared to the traditional method. Apart from there being no meaningful difference, an improvement has been observed for the benefit of Experimental Group among the averages in terms of willingness and negative attitude factors when the averages have been examined together. Accordingly, it can be said that there has been a development for the benefit of the Experimental Group in the factors of willingness and negative attitude although the teaching using Lego Mindstorms Ev3 does not have a meaningful effect on the students' attitudes towards teaching of programming according to the traditional method. When the post test average points of the students were examined, it has been seen that the average of willingness factor is 75.8, the average of negative attitude is 78.8 and average of necessity factor is 32.2. Thus, it can be said that students consider the programming education as unnecessary and negative perception is pretty high although the levels of willingness are also high. Accordingly, the fact that the attitudes of the students both in test and control group are largely negative might have caused the meaningful differentiation to occur.

When Table 8 was examined, it has been observed that Lego Mindstorms Ev3 design activities differentiate ($t_{(2-51)}$=-4.235; p<0.05) the academic success towards C++ of the students at a meaningful level compared to the traditional method. When the averages were analyzed, it has been seen that the average academic success of the Experimental Group is 38.9, the average of the control group is 14.7 and the meaningful differentiation is for the benefit of the Experimental Group. Accordingly, it can be said that teaching using the Lego Mindstorms Ev3 design activities contributes more meaningfully to the academic success of the students as regards C++ programming language than the traditional method.

When Table 8 was examined, it has been seen that Lego Mindstorms Ev3 design activities have not differentiated the self-efficacy beliefs of the students towards C++ compared to the traditional method ($t_{(2-51)}$=-0.212; $p<0.05$). Accordingly, it can be said that teaching using Lego Mindstorms Ev3 design activities is not effective in self-efficacy of the students towards C++ programming compared to the traditional method.

4. Discussion and Conclusion

The average academic success of the students as regards the C++ programming is generally at a medium level. However, it can be said that this level does not comprise ability. This situation might have been caused by the fact that the students haven't been supported enough in high level thinking abilities, they consider C++ programming irrelevant for their occupation, their attitudes towards learning is low and self-efficacy beliefs are weak. Although the average academic success as regards programming is at the medium level, a great part of the students see themselves as highly competent at C++ programming. It is an interesting contradiction. Students' academic success might be affected negatively by the fact that they consider the insufficient programming skills they have as sufficient. A great part of the students are willing to learn C++. However, an important part of the students have a negative attitude towards learning C++. In addition, it can be said that most of these students find it unnecessary to learn C++. Moreover, these results do not differ among the male and female students. Consequently, students seem to be reluctant to get more programming education as they consider the programming skills they have as sufficient. As a result, their academic success is low. It is possible to say that traditional education of programming is maintained most of the time by explaining code structures and solving small examples. At the end of an education like that, students can achieve a sufficient level when they are asked about subjects like the register and usage of commands. However, many problems might occur in academic success when they are asked questions like the solution of a problem, an innovative algorithm or logical errors in an algorithm. Since programming is a complex process, high level thinking abilities should play a role (Korkmaz and Altun, 2014; Fang, 2012; Korkmaz, 2012; Lau andYuen, 2009; Wang et al, 2012). Therefore, the contradiction mentioned above has occurred; the students have negative attitudes although they are willing and accordingly their academic success might have been low.

Genders of the students are not effective in the students' academic success, attitudes towards learning programming and the levels of self-efficacy beliefs. Yağcı (2016) stated that there is a plethora of computing or programming features on which gender is not effective.

Apart from the fact that Lego Mindstorms Ev3 does not have any meaningful effect on the attitudes of the students towards teaching programming compared to the traditional method, a development has occurred for the benefit of the Experimental Group in the factors of willingness and negative attitude. It is possible to encounter much evidence proving that Lego robot applications have positive effects on attitude. The agreed point of all the studies is that students and teachers have developed positive attitudes in the competitions they attend and in the courses after school even if they had no experience before (Seddighin, 2013; Vollstedt, 2005; Webb, 2013; Yalçın, 2012).

In addition, it is possible to see much evidence in the literature proving that Lego increases problem solving skills (Gibbon, 2007; Lenamond, 1991; McDaniel, 2004). When the post test average points were examined, the average of the willingness factor is 75.8, the average of negative attitude is 78.8 and the average of necessity factor is 32.2. Although the students' levels of willingness are high, it has been seen that they see the programming education as unnecessary and they have negative attitudes towards it. Accordingly, the meaningful differentiation might have resulted from the negative attitudes of the students both in the test and the control group. As stated before, since the programming process is closely related to high level thinking abilities, this education gets even harder. Especially for the students whose high level thinking abilities haven't been supported enough, programming courses might be a nightmare. They might be affected negatively by this situation. It is possible to see much evidence in the literature suggesting that attitudes towards programming are an important problem (Anastasiadou and Karakos, 2011; Erdogan et al., 2008; Sacks et al., 1993.)

Furthermore, it has been identified that education using Lego Mindstorms Ev3 design activities does not have any effect on self-efficacy of the students for C++ programming language compared to the traditional method. As mentioned above, the differences might have resulted from the high self-efficacy of the students both in the test and the control group towards programming.

The education using Lego Mindstorms Ev3 design activities contributed more meaningfully to the academic success of the students in terms of C++ programming language compared to the traditional method. This finding shows coherence with the literature. An increase has been determined in the post test points of the students in the experimental studies for the benefit of the Experimental Group (Özdoğru, 2013). Similarly, it has been seen that the students attending robotic courses have higher academic performances than the students not attending the course (Koumoullos, 2013). A researcher (Tse, 2009) has organized a control kit in postgraduate thesis showing that students lack the ability to put into practice their theoretical knowledge. The study was conducted with undergraduate students using the questionnaire and quantitative method. Almost all of the participants have expressed that studying with physical models makes it easier to understand theoretical information. In the end, abstract knowledge has become more comprehensible and the students have learned real life experiences by creating and having fun. Howell (2012) has developed a robotic system that is cheap, flexible and that can be used for educational purposes in the doctorate thesis. He applied it to physics and computer programming in high school and computer sciences and bioengineering in university. Most of the students who have been asked their opinion about pre and post tests in the study conducted by the quantitative method have stated that using robotics in class helps materialize abstract concepts and is fun and motivating.

As a consequence, it can be proposed that the Lego Mindstorms Ev3 design activities be used in the programming courses in the bachelor's degree course for the purpose of improving the students' algorithmic thinking and programming skills.

Acknowledgement

This research has been executed within the scope of SEB-BAP 14-37 project endorsed by Amasya University Project Coordination Unit of Scientific Researches.

References

Anastasiadou, S.D., Karakos, A.S. (2011). The beliefs of electrical and computer engineering students' regarding computer programming. *The International Journal of Technology,Knowledge and Society,* 7(1): 37-51.

Czerkawski, B., Lyman, E. (2015). Exploring Issues about Computational Thinking in Higher Education. *TechTrends*, 59(2), 57-65.

Çayır, E. (2010). *Assignment the effect of learning environment supported by lego-logo on science process skill and self concept*, Master dissertation, Sakarya University, Sakarya.

Erdogan, Y., Aydin, E., Kabaca, Y.T. (2008) Exploring the Psychological Predictors of Programming Achievement. *Journal of Instructional Psychology*, 35(3): 264-270.

Fang, X. (2012). Application of the participatory method to the computer fundamentals course, Affective Computing and Intelligent Interaction. *Advances in Intelligent and Soft Computing,* 137: 185-189.

George, D., Mallery, M. (2010). *SPSS for Windows Step by Step: A Simple Guide and Reference,* 17.0 update (10a ed.) Boston: Pearson

Gibbon, L.W. (2007). *Effects of Lego Mindstorms on Convergent and Divergent Problem Solving and Spatial Abilities in Fifth and Sixth Grade Students*. Doctoral dissertation, Seattle Pacific University, Washington, USA.

Gomes, A., Mendes, A., J.(2007). Learning to program-difficulties and solutions. *International Conference on Engineering Education, ICEE*.

Güntürkün, E. (2009*). Historical and structural development of construction toys (Using the example of LEGO company)*, Master dissertation, Marmara University, İstanbul.

Howell, A. (2012). *Development and Validation of A Low Cost, Flexible, Open Source Robot For Use As A Teaching and Research Tool Across The Educational Spectrum*. Doctoral dissertation, Binghamton University, New York, USA.

Huizinga, J. (2013). *Homo Ludens Oyunun Toplumsal İşlevi Üzerine Bir Derleme* (Çev. M. A. Kılıçbay) (4. baskı). İstanbul,(Eserin aslının yayın tarihi 1955). Ayrıntı Yayıncılık.

Hwang, G., Wu, P., Chen, C. (2012). An online game approach for improving students' learning performance in web-based problem-solving activities. *Computers and Education*, 59: 1246–1256. doi:10.1016/j.compedu.2012.05.009.

Jenkins, T. (2002). On the difficulty of learning to program. *in Proc. of the 3 rd Annu. LTSN_ICS Conf.,* Loughborough University, United Kingdom, pp. 53-58.

Katai, Z., Juhasz, K., and Adorjani, A. K. (2008). On the role of senses in education. *Computers and Education*, 51(4), 1707–1717. doi:10.1016/j.compedu.2008.05.002.

Korkmaz, Ö. (2012). The Impact of Critical Thinking and Logical-Mathematical Intelligence on Algorithmic Design Skills. *Journal of Educational Computing Research*, 46(2):173-193. DOI: 10.2190/EC.46.2.d.

Korkmaz, Ö. (2013). Students' Difficulties in and Opinions about Designing Algorithms According to Different Instructional Applications. *Energy Education Science and Technology Part B: Social and Educational Studies* , 5(1).209-218.

Korkmaz, Ö., Altun, H. (2014a). A Validity and reliability study of the Learning Computer Programming Attitude Scale (LeCoPAS). *Mevlana International Journal of Education*, 4(1): 30-43 http://dx.doi.org/10.13054/mije.13.73.4.1

Korkmaz. Ö., Altun, H. (2014b). Adapting Computer Programming Self-Efficacy Scale and Engineering Students' Self-Efficacy Perceptions. *Participatory Educational Research (PER).* Vol. 1(1), pp. 20-31.

Kotaman , H. (2008). Self-Efficay Belif and Enhancement of Learning Performance. *Uludağ University The Journal of Educational Faculty*, XXI (1), 111-133.

Koumoullos, M.G. (2013). *The Academic Differences Between Students Involved in School – Based Robotics Programs and Students Not Involved in School- Based Robotic Programs.* Doctoral dissertation, St. John University, New York, USA.

Lai, C., Wang, Q. and Lei, J. (2012). What factors predict undergraduate students' use of technology for learning? A case from Hong Kong. *Computers and Education*, 59(2): 569–579. doi:10.1016/j.compedu.2012.03.006.

Lau, W.W.F., Yuen, A.H.K. (2011). Modelling programming performance: Beyond the influence of learner characteristics. *Computers and Education*, 57: 1202–1213. doi:10.1016/j.compedu.2011.01.002

Lenamond, D.B. (1991). *A Comparison of the Effects of Lego Tc Logo and Problem Solving Software on Problem Solving Skills*. Master dissertation, University of Houston- Clear Lake, Houston, Texas, USA

McDaniel, A.R. (2004). *How A Group of Elementary Deaf Students Interact with LEGO LOGO Activities: A Case Study*. Doctoral dissertation, The University of Tennessee, Knoxville, USA.

Milne, I. and Rowe, G. (2002). Difficulties in Learning and Teaching Programming—Views of Students and Tutors. *Education and Information Technologies*, 2002, 7(1): 55-66.

Moore, V.S. (1999). Robotics: Design through Geometry. Technology Teacher, 59(3), 17-22.

Özdoğru, E. (2013). *The effect of Lego programme based science and technology education on the students academic achievement, science process skills and their attitudes toward Science and Technology course for pyhsical facts learning field*. Master dissertation, Dokuz Eylül University Educatonal Science Institute, İzmir.

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

Prensky, M. (2003). Digital game-based learning. *Computers in Entertainment (CIE),* 1(1), 21-21.

Ramalingam, V., and 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.

Sacks, C., Bellisimo, Y. and Mergendoller, J (1993) Attitudes toward computers and computer use: the issue of gender. *Journal of Research on Computing in Education*, 26: 257-269.

Seddighin, S. (2013). *Evaluation of a Professional Development Workshop on Integration of Robotics into Early Childhood Classrooms*. Master dissertation, Tufts University, Medford, Massachusetts, USA.

Tabachnick, B.G. and Fidell L.S. (2013). *Using Multivariate Statistics (sixth ed.).* Pearson, Boston

Tan, P., Ting, C., Ling, S. (2009). Learning Difficulties in Programming Courses: Undergraduates' Perspective and Perception. International *Conference on Computer Technology and Development*,42-46. doi:10.1109/ICCTD.2009.188

Tse, S.B. (2009). *MINDSTORMS Controls ToolKit: Hands-on, Project- Based Learning of Controls*. Master dissertation, Tufts University, Medford, Massachusetts, USA.

Varnado, T. E. (2005). *The Effects of a Technological Problem Solving Activity on FIRST LEGO League Participants' Problem Solving Style and Performance*. Doctoral dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Varney, M. W., Janoudi, A., Aslam, D. M. ve Graham, D. (2012). Building Young Engineers: TASEM for Third Graders in Woodcreek Magnet Elementary School. *IEEE Transactions on Education,* 55(1).

Vollstedt, A.M. (2005). *Using Robotics to Increase Student Knowledge and Interest in Science, Technology, Engineering and Math*. Master dissertation, University of Nevada, Reno, USA.

Wang, Y., Li, H., Feng, Y., Jiang, Y., and Liu, Y. (2012). Assessment of programming language learning based on peer code review model: Implementation and experience report. *Computers and Education*, 59: 412–422. doi:10.1016/j.compedu.2012.01.007.

Webb, H. C. (2013*). Injecting Computational Thinking Into Computing Activities For Middle School Girls.* Doctoral dissertation, Information Sciences and Technology, The Pennsylvania State University, USA.

Yağcı, M. (2016). Effect of attitudes of information technologies (IT) preservice teachers and computer programming (CP) students toward programming on their perception regarding their self-sufficiency for programming. *International Journal of Human Science* 13(1), 1418-1432, doi:10.14687/ijhs.v13i1.3502.

Zaharijaa, G., Mladenovića, S., and Boljat, I. (2013). Introducing the Basic Programming Concepts to Elemantary School Children. *4 th International Conference on New Horizons in Education, Social and Behavioral Sciences*, 106, 1576-1584.

Received July 11, 2016, revised August 23, 2016, accepted September 18, 2016