



European Journal of Educational Research

Volume 7, Issue 4, 925 - 933

ISSN: 2165-8714

<http://www.eu-jer.com/>

Gender Differences in Middle School Students' Attitudes and Self-Efficacy Perceptions towards mBlock Programming *

Mustafa Serkan Gunbatar **
Van Yuzuncu Yil University, TURKEY

Halit Karalar
Mugla Sitki Kocman University, TURKEY

Received: August 14, 2018 • Revised: October 5, 2018 • Accepted: October 5, 2018

Abstract: This study aims to assess the effects of teaching programming with mBlock on self-efficacy perceptions and attitudes considering programming. Particularly, this study tries to research whether there is a gender difference in middle school students or not. The study was conducted in pre-test/post-test quasi experimental design. The participants of the study which was completed in twelve weeks were 82 middle school students. The data were collected through "Educational Computer Games Assisted Learning Coding Attitude Scale" and "Computer Programming Self-efficacy Scale". The results of the research indicate that although the self-efficacy perceptions of boys towards programming were higher than the girls' at the beginning of the research, this difference was closed at the end of the research. The results also show that teaching programming with mBlock to middle school students did not cause gender differences in self-efficacy perceptions and attitudes regarding programming. Although girls' attitudes regarding programming were slightly higher than boys', the difference was not considered to be significant. In addition, it was found that programming with mBlock significantly increased students' self-efficacy perceptions and attitudes towards programming. As a result, teaching programming with mBlock can provide similar possibilities for both genders in self-efficacy perceptions and attitudes regarding programming.

Keywords: *mBlock, teaching programming, middle school students, attitudes towards programming, self-efficacy perceptions towards programming.*

To cite this article: 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), 923-933. doi: 10.12973/eu-jer.7.4.923

Introduction

Since programming has been used to improve students' problem solving, systematic thinking, critical thinking, algorithmic thinking, creative thinking and design skills (Clements & Gullo, 1984; diSessa, 2000; Fessakis, Gouli, & Mavroudi, 2013; Liao & Bright, 1991), it is important for students to be taught programming from very early ages. However, despite the increasing importance of programming, it is difficult for students to learn programming because it is difficult and tedious for them to learn programming and many obstacles are encountered in teaching programming.

Much research has been done in the field about obstacles encountered in teaching programming. These obstacles can be classified as external and internal obstacles. External obstacles are the ones that are not related with students, such as the inappropriate teaching methods used by teachers (Guzdial & Soloway, 2002; Lee, 2011); the use of text-based programming languages and the necessity of knowing the syntax and commands of these languages (Jenkins, 2002); the abstractness of the basic concepts of programming; the difficulty of testing the program and correcting errors (Pausch, Dann, & Cooper, 2000). On the other hand, internal obstacles are the ones that are directly related to the learners themselves, such as the negative attitudes towards programming (Courte, Howard, & Bishop-Clark, 2006; Cetin & Ozden, 2015; Korkmaz & Altun, 2013; Yukselturk & Altioek, 2017) and low self-efficacy perceptions (Altun & Mazman, 2012; Askar & Davenport, 2009; Yukselturk & Altioek, 2017).

According to Bandura (1997), self-efficacy perception is an important factor in predicting an individual's behavior. He defines self-efficacy as the individual's own judgment on the capacity of the individual to achieve success with organizing activities, can affect an individual's relative involvement, the effort he or she has spent to accomplish that task, and the ambition that will be shown to complete the task. Even if the individual has enough knowledge and skills to perform a particular task; success, course and career choices can be negatively affected if he / she has doubts about

* Early version of the study was presented at 2 nd International Symposium of Limitless Education and Research (ISLER 2018)

** Corresponding author:

Mustafa Serkan Gunbatar , Van Yuzuncu Yil University, Department of Computer Education and Instructional Technologies, Van- Turkey.
Email: mustafaserkan@yyu.edu.tr

his performance (Altun & Mazman, 2012; Baser, 2013). Consequently, self-efficacy perceptions towards programming affect programming success (Mazman & Altun, 2012) and programming performance (Altun, & Mazman, 2015).

The other critical factor determining students' perceptions of computer programming is the attitudes of students. Students' attitudes toward programming can affect their success in computer science courses and career choices (Cetin & Ozden, 2015). Studies have shown that boys' attitudes towards programming are significantly higher than girls' (Baser, 2013, Korkmaz & Altun, 2013, Rubio, Romero-Zaliz, Mañoso & de Madrid, 2015) and there is a positive relationship between attitude towards programming and programming achievement (Baser, 2013).

The number of students making a career in computer science is gradually decreasing (Baser, 2003; Chen et al., 2017), and the vast majority of these students are male, despite the increasing need for IT professionals today (Chen et al., 2017; Grover & Pea, 2013). Although there is no significant difference between the programming performances of students and their gender (Lau & Yuen, 2011; Su, Yang, Hwang, Huang, & Tern, 2014), there are differences in the perception of programming for girls and boys (Rubio et al., 2015). These gender based differences in students' self-efficacy perceptions and attitudes towards programming may affect their interests in computer science and career choice.

One of the different approaches used to provide girls with the opportunity to love programming to develop positive attitudes towards programming, and to improve self-efficacy perceptions is the use of visual programming environments (Denner, Werner, & Ortiz, 2012; Monroy-Hernández & Resnick, 2008). The abstract concepts of programming become concrete through the block based graphical programming environments which are developed by considering the level of development of the students. Students can see how the commands work, instantly test the ideas and get the feedback, detect the errors more easily, learn by trial and error (Kelleher & Pausch, 2005, Lye & Koh, 2014). Through these settings, students can focus only on designing and creating (Grover & Pea, 2013). Students can learn actively in such a learning environment by constructing their new learning (Bruner, 1960). Alice and Scratch are examples of these environments.

Scratch is a programming environment which is free of charge. This environment was developed at the MIT Media Laboratory by Lifelong Kindergarten group. The Scratch includes more than 100 programming blocks under nine categories. It reinforces young learners to produce personal games, simulations, and interactive stories, then share these productions in an online platform. According to Scratch web page (Scratch, 2018), it is designed especially for ages 8 to 16, available in more than 150 different countries with more than 40 languages.

According to Resnick (2002), today's technology readers who learn programming with Scratch become technology writers by learning the language of technology and become able to produce with technology, and learn design skills at the same time. At present, the number of users registered on Scratch is more than 28 million and the number of projects shared is more than 31 million. The popularity of Scratch has led to the development of different environments that are based on Scratch. As one of these environments, mBlock is a visual programming environment developed on the basis of Scratch 2.0 and offering the possibility of programming robotic and micro-controller programming in addition to Scratch 2.0.

Studies on Scratch show that it is effective in removing many obstacles to programming instruction (Grover, Pea, & Cooper, 2015; Meerbaum-Salant, Armoni, & Ben-Ari, 2013; Sáez-López, Román-González, & Vázquez-Cano, 2016). Similarly, there are studies (Yukselturk & Altiok, 2017) that indicate that Scratch is also effective in overcoming obstacles such as negative attitude towards programming and low self-efficacy. However, the impact of the mBlock environment built on Scratch on middle school students is not well known. Moreover, it has not been adequately studied how the self-efficacy perceptions and attitudes towards programming of middle school students are changed according to their genders. If a high self-efficacy perception and positive attitude towards programming among girls at middle school level can be developed, it can be ensured that girls may make career choices in the field of computer science. Hence, this study aims to assess the effects of teaching programming with mBlock on self-efficacy perceptions and attitudes considering programming. Particularly, this study tries to research if there is a gender difference in middle school students' self-efficacy perceptions and attitudes regarding programming. Therefore, this study addresses the following research questions:

- What is the effect of mBlock programming on middle school students' self-efficacy perceptions towards programming?
- What is the effect of mBlock programming on middle school students' attitudes towards programming?
- Is there a gender difference on middle school students' self-efficacy perceptions towards programming?
- Is there a gender difference on middle school students' attitudes towards programming?

Methodology

Research Design

The study was conducted in pre-test/post-test quasi experimental design that lacks the element of random assignment (Fraenkel & Wallen, 2009). This study used methodological design to compare the extend of change in students' self-

efficacy perceptions and attitudes towards programming. The independent variable of the study was gender. The dependent variables of the study were self-efficacy perceptions and attitudes towards programming. The scores of attitudes towards programming were gathered through the “Educational Computer Games Assisted Learning Coding Attitude Scale (ECGALCAS)” and of self-efficacy perceptions towards programming were gathered through the “Computer Programming Self-efficacy Scale (CPSES)”. The design of the study was presented in Table 1.

Table 1. Design of the study

N	Pre-test	Treatment (12 weeks)	Post-test
82	ECGALCAS CPSES	T	ECGALCAS CPSES

ECGALCAS: Educational Computer Games Assisted Learning Coding Attitude Scale

CPSES: Computer Programming Self-efficacy Scale

Participants

The research was conducted in a public middle school in a big city of eastern Turkey. Like all middle schools in Turkey, it covers students among 5, 6, 7, and 8 grades. The participants were 82 middle school students (39 girls and 43 boys; mean age of 13) from sixth grade in three different classes attending an information technology course. The students have not prior programming experiences.

Procedure

Teaching programming with mBlock continued for a period of twelve weeks. In this study, mBlock version 3 was chosen for programming teaching because students were given basic programming skills. The procedure was maintained for two hours per week and lasted 24 hours in total. In order for the further research and implementations, instructions and necessary competencies for smart devices development with Arduino will be given to students in the following teaching. The fundamental purpose of this future plan is to provide students to use the same learning environment.

Before the process, managers, parents, teachers and students were informed about the study. The necessary permissions were taken from the parents and managers. The volunteer students participated in this research. The Information Technologies teacher of the school and the researchers developed a twelve-week course outline and implemented accordingly. In the first week, students were given information about software and problem concepts. It was also pointed out that computers could solve their problems by applying the instructions step by step. Examples of daily problem cases and solution steps were shown. Then, the scales for attitudes and self-efficacy perceptions towards programming were administrated to the students.

In the second week, the importance of creating an algorithm was mentioned. Algorithms for solution of example problems were introduced. The concept of flow chart was explained, showing the locations and purposes of geometric shapes in flow chart. Implementations about the flow diagram representation of example algorithms were conducted.

In the third week, the characters and Events blocks in the mBlock application were introduced. Control and Looks were introduced. The pupils performed name print activities with puppets and moved their names with simple commands (zoom in, increase color effect, fish eye etc.).

In the fourth week, cut, copy, enlarge etc. commands were explained on the screen. Coordinate plane was introduced. Motion blocks, direction concepts and related commands were explained. The stage section was revealed.

In the fifth week, the students made a character with mBlock, moved it and made different studies about the appearance of the character (Figure 1).

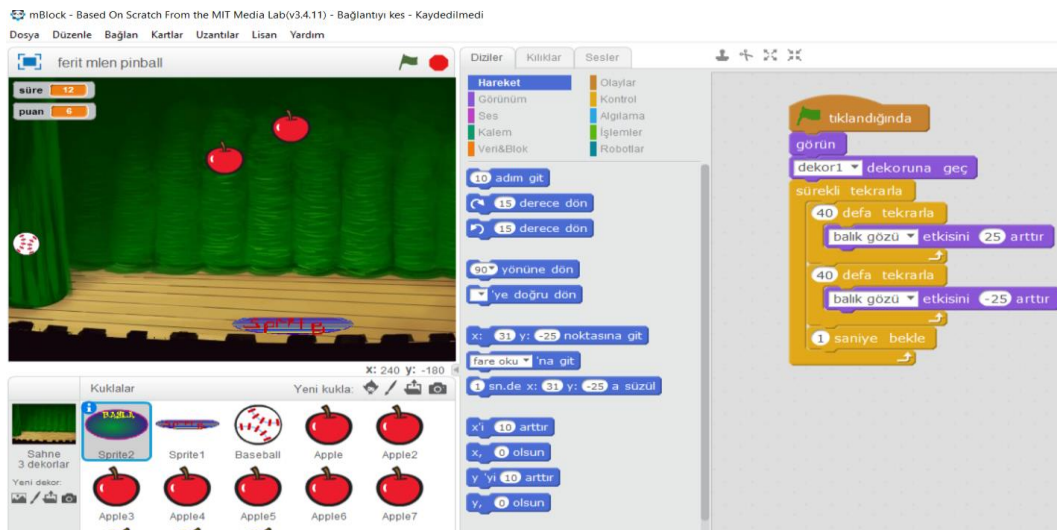


Figure 1. Screenshot of creating and moving characters

In the sixth week, sample aquarium application was shown to the students. The Motion blocks commands showed how the fish moved. Students were asked to practice the aquarium. Before application, students created an algorithm with various questions. With this application, the commands in the Events, Control, Motion, and Looks blocks were actively used (Figure 2).

In the seventh week, Labyrinth was applied with the help of commands in the Events, Control, Motion, Looks and Sensing blocks. Thanks to this application, the commands in the blocks were used in a mixed way.

In the eighth week, the example Pinball application was shown. It was revealed how the characters move, what the characters do when they touch each other. They were asked to do a similar pinball application.

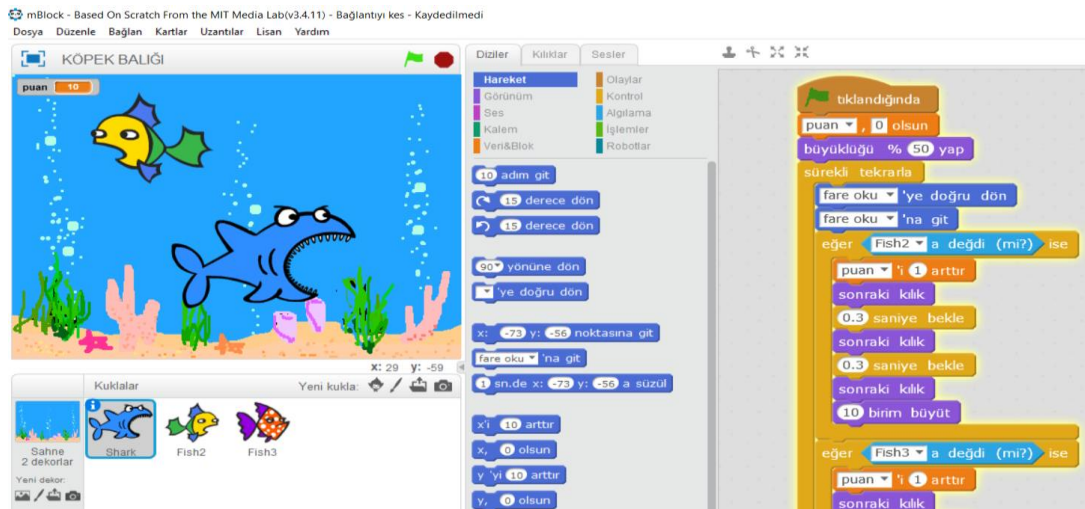


Figure 2. Screenshot of the aquarium application

In the ninth week, Operators block commands were shown. Small samples were presented. Later on, a question bank consisting of 5 Questions was formed with the commands shown. The students were asked to develop a 15-question question bank (Figure 3).



Figure 3. Screenshot of question bank application

In the tenth week, the concept of the variable was explained, the commands in the Data blocks were shown. Small samples were presented, then students were asked to practice catching fish. They wanted the user to show the scores they got with the variables they created in their practice and the points they lost.

In the eleventh week, the Pen blocks was introduced and related commands were shown. Small applications were conducted. Later, students were asked to draw different geometric shapes.

In the twelfth week, a table hockey practice was shown. The students wanted to play this game in cooperation. Thanks to this game, commands for Events, Control, Motion, Looks, Sensing, Operations and Data blocks were repeated. Then, the scales for self-efficacy perceptions and attitudes towards programming were re-administrated to the students as post-tests.

Data collection tools

Educational Computer Games Assisted Learning Coding Attitude Scale (ECGALCAS) developed by Kececi, Alan, & Kirbag-Zengin (2016) was used to examine students' self-efficacy perceptions towards programming. Being a 5-point Likert-type scale, it consists of 28 items falling under three factors, *desire for coding learning, interest in the use of computer games for educational purposes in classes, and concerns about computer a socialization*. The reliability of the scale was .83. The value calculated in this study was .87.

Computer Programming Self-efficacy Scale (CPSES) developed by Kukul, Gokcearslan & Gunbatar (2017) was used to examine students' attitudes towards programming. Being a 5-point Likert-type scale, it consists of 31 items falling under one factor. The reliability of the scale was .95. Similarly, the value calculated in this study was .96.

Data analysis

Before analysis, normality of difference between pre-test and post-test scores of CPSES and ECGALCAS were tested. According to the Kolmogorov-Smirnov test, the data for the pre-tests and post-tests ($p > .05$) were distributed normally. Similarly, normality of pre-test and post-test scores according to gender of CPSES and ECGALCAS were tested. According to the Shapiro-Wilk test, the data for the pre-tests and post-tests were distributed normally ($p > .05$). After testing the assumptions, the data obtained in the study were analyzed with descriptive statistics, t-test and one-way ANCOVA test.

Results

Effect of mBlock programming on middle school students' self-efficacy perceptions towards programming

Table 2. t-test results of pre-test and post-test of CPSES

Measurement	N	Mean	Std	df	t	p	η^2
Pre-test	82	1.40	0.54	81	-13.15	0.00*	0.68
Post-test	82	2.50	0.67				

* $p < 0.05$

According to results of paired samples t-test for pre-test and post-test of CPSES (Table 2), there found a significant difference between pre-test and post-test result of students' self-efficacy perceptions towards programming [$t_{(81)}=-13.15, p<.05$]. The effect size for this significant difference was large. This result may show that programming via mBlock may have an effect on students' self-efficacy perceptions towards programming.

Effect of mBlock programming on middle school students' attitudes towards programming

Table 3. t-test results of pre-test and post-test of ECGALCAS

Measurement	N	Mean	Std	df	t	p	η^2
Pre-test	82	1.98	0.43				
Post-test	82	2.46	0.61	81	-7.76	0.000*	0.43

* $p<0.05$

According to these result of paired samples t-test for pre-test and post-test of ECGALCAS (Table 3), there were significant difference between pre-test and post-test result of students' attitudes towards programming [$t_{(81)}=-7.76, p<.05$]. The effect size for this significant difference was large. This result may show that programming via mBlock may have an effect on students' attitudes towards programming.

Gender difference on middle school students' self-efficacy perceptions towards programming

Table 4. t-test results of pre-test and post-test according to gender of CPSES

Measurement	Gender	N	Mean	Std	df	t	p
Pre-test	Girl	39	1.23	0.55	80	-2.682	0.009*
	Boy	43	1.54	0.48			
Post-test	Girl	39	2.43	0.63	80	-0.931	0.355
	Boy	43	2.57	0.70			

* $p<0.05$

According to results of independent t-test according to gender for pre-test and post-test of CPSES (Table 4), there was a significant difference between pre-test scores of self-efficacy perceptions towards programming [$t_{(80)}=-2.68, p<.05$] in favor of boys. When the analysis results of post-test scores of students according to gender were examined, no significant difference was found [$t_{(80)}=-0.93, p>.05$].

Since the groups were not equal in the sense that there was a difference between the self-efficacy perceptions of the students before the experiment, the change in the self-efficacy perceptions of the students was examined with the one-way ANCOVA test by controlling the pre-test scores of the students (Table 5).

Table 5. One Way ANCOVA Analysis result

Source of Variance	Sum of Squares	df	Mean Square	F	p
Self-efficacy pre-test	1.31	1	1.31	3.01	0.09
Gender	0.07	1	0.07	0.16	0.69
Error	34.26	79	0.43		
Total	548.65	82			

When the pre-test scores of the students were controlled, no significant difference was found [$F_{(1,80)}=0.16, p>.05$]. This result may show that programming via mBlock may close the gap between self-efficacy perceptions towards programming against girls.

Gender difference on middle school students' attitudes towards programming

Table 6. t-test results of pre-test and post-test according to gender of ECGALCAS

Measurement	Gender	N	Mean	Std	df	t	p
Pretest	Girl	39	1.98	0.51	80	-0.05	0.96
	Boy	43	1.98	0.36			
Posttest	Girl	39	2.52	0.58	80	0.87	0.39
	Boy	43	2.40	0.63			

According to result of independent t-test according to gender for pre-test and post-test of ECGALCAS (Table 6), no significant gender difference was found between students' pre-test scores of attitudes towards programming [$t_{(80)} = -0.05, p > .05$]. When the analysis results of post-test scores were examined, no significant gender difference was found [$t_{(80)} = 0.87, p > .05$]. This result may show that programming with mBlock may offer equal opportunities for girls and boys in terms of attitudes towards programming.

Discussion and Conclusion

This study aims to assess the effects of teaching programming with mBlock on self-efficacy perceptions and attitudes considering programming. Particularly, this study tries to research whether there is a gender difference in middle school students' self-efficacy perceptions and attitudes regarding of programming.

The results of the study showed that teaching programming with mBlock increased middle school students' self-efficacy perceptions and attitudes regarding programming significantly. The finding is similarly observed in several study findings: In Saes-Lopez et al. (2016) study investigating the use of Scratch among 5th to 6th graders in Spain, they found high and positive attitudes and motivation depending on the use of Scratch. Similarly, Kalelioglu & Gulbahar (2014) studied the effects of Scratch programming on problem solving skills and attitudes, and as a result, it is highlighted that primary school students displayed positive attitudes toward programming. Kelelioglu (2015) investigated the effect of teaching code.org site and found that students developed a positive attitude towards programming. In relation with the student based studies, Yukselturk & Altioek (2017) analyzed the effects of programming with Scratch on the views of preservice teachers about computer programming and they concluded that participants generally had higher and more positive self-efficacy perceptions and attitudes. All these results commonly embark that visual programming environments can increase students' self-efficacy perceptions and attitudes toward programming.

When the self-efficacy perception scores of the students were examined, the results of the research showed that the boys' pre-test self-efficacy perception regarding programming was significantly higher than girls'. Although the reason for this is not fully explained, we think that this difference may be caused by previous programming experience. According to Rubio et al. (2015), programming experience is influential on the perceptions of students; for males and females who do not have programming experience, the perception of males is higher than females and men find it easier to program. Askar & Devenport (2009) studied the self-efficacy for Java programming among first year engineering students and the results showed that male students had higher self-efficacy when compared to females'. This gender gap may not change when traditional text-based programming environments are used. When the post-test self-efficacy perception scores of the students were examined, no significant gender difference was found. This result may show that programming with mBlock may close the gap between self-efficacy perceptions towards programming against girls.

When the pre-test and post-test attitude scores of the students were examined, the results of the study showed that no significant gender difference was found. The finding is similarly observed in several study findings: Zuckerman, Blao & Monroy-Hernandez (2009) studied Scratch in terms of different variables, however, gender variable had no significant value in the findings. Similarly, Cetin (2016) investigated the effect of Scratch programming on preservice teachers' understanding of fundamental programming concepts and their attitudes toward programming and obtained no significant gender difference in attitudes toward programming. Additionally, Su et al. (2014) evaluated Scratch programming in an elementary school in northern Taiwan to analyze the effects of Scratch on programming performance and no gender difference was found. Consistent with previous studies, there was no significant difference in students' self-efficacy perceptions and attitudes towards programming in this study. The results show that when teaching programming through visual programming environments, the gap between gender differences can be closed in terms of many variables.

This study is primarily limited by eighty-one 6th grade students who were consisted from three different classes, 12 weeks duration and activities prepared by the IT teacher. This study needs to be confirmed using data from other degrees in different schools. If these results can be reproduced in other schools, it could be increase the number of girls interested in computer science.

References

- Altun, A., & Mazman, S. G. (2012). Programlamaya ilişkin öz yeterlilik algısı ölçeğinin Türkçe formunun geçerlilik ve güvenilirlik çalışması [The validity and reliability study of the Turkish version of the self-efficacy perception regarding programming scale]. *Journal of Measurement and Evaluation in Education and Psychology*, 3(2), 297–308.
- Altun, A., & Mazman, S. G. (2015). Identifying latent patterns in undergraduate students' programming profiles. *Smart Learning Environments*, 2(13), 1-16. doi:10.1186/s40561-015-0020-0
- Askar, P., & Davenport, D. (2009). An investigation of factors related to self-efficacy for JAVA programming among

- engineering students. *The Turkish Online Journal of Educational Technology*, 8(1), 26–32.
- Bandura, A. (1977). Self-efficacy: toward a unifying theory of behavioral change. *Psychological Review*, 84(2), 191–215. doi:10.1037/0033-295X.84.2.191
- Baser, M. (2013). Attitude, gender and achievement in computer programming. *Middle-East Journal of Scientific Research*, 14(2), 248–255. doi:10.5829/idosi.mejsr.2013.14.2.2007
- Bruner, J. (1960). *The process of education*. Cambridge: Harvard University Press.
- Cetin, I., & Ozden, M. Y. (2015). Development of computer programming attitude scale for university students. *Computer Applications in Engineering Education*, 23(5), 667–672. doi:10.1002/cae.21639
- Cetin, I. (2016). Preservice teachers' introduction to computing: Exploring utilization of Scratch. *Journal of Educational Computing Research*, 54(7), 997–1021. doi:10.1177/0735633116642774
- 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. doi:10.1016/j.compedu.2017.03.001
- Clements, D. H., & Gullo, D. F. (1984). Effects of computer programming on young children's cognition. *Journal of Educational Psychology*, 76(6), 1051–1058. doi:10.1037/0022-0663.76.6.1051
- Courte, J., Howard, E. V., & Bishop-Clark, C. (2006). Using Alice in a computer science survey course. *Information Systems Education Journal*, 4(87), 3–7.
- Denner, J., Werner, L., & Ortiz, E. (2012). Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts? *Computers & Education*, 58(1), 240–249. doi:10.1016/j.compedu.2011.08.006
- diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge, MA: MIT Press.
- Fessakis, G., Gouli, E., & Mavroudi, E. (2013). Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. *Computers & Education*, 63, 87–97. doi:10.1016/j.compedu.2012.11.016
- Fraenkel, J. R., & Wallen, N. E. (2009). *How to design and evaluate research in education* (7th. ed.). NY: McGraw-Hill.
- Grover, S., & Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1), 38–43. doi:10.3102/0013189X12463051
- Grover, S., Pea, R., & Cooper, S. (2015). Designing for deeper learning in a blended computer science course for middle school students. *Computer Science Education*, 25(2), 199–237. doi:10.1080/08993408.2015.1033142
- Guzdial, M., & Soloway, E. (2002). Teaching the Nintendo generation to program. *Communications of the ACM*, 45(4), 17. doi:10.1145/505248.505261
- Jenkins, T. (2002). On the difficulty of learning to program. In *Third Annual Conference of the LTSN-ICS* (pp. 53–58). UK: Loughborough University.
- Kalelioglu, F., & Gulbahar, Y. (2014). The effect of teaching programming via scratch on problem solving skills: A discussion from learners' perspective. *Informatics in Education*, 13(1), 33–50.
- Kalelioglu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, 52, 200–210. doi:10.1016/j.chb.2015.05.047
- Kececi, G., Alan, B., & Kirbag Zengin, F. (2016). Educational computer games assisted learning coding attitude scale: Validity and reliability study. *Education Sciences (NWSAES)*, 11(3), 184–194. doi:10.12739/NWSA.2016.11.3.1C0661
- Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A survey of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR)*, 37(2), 83–137. doi:10.1145/1089733.1089734
- Korkmaz, O., & Altun, H. (2013). Engineering and CEIT student's attitude towards learning computer programming. *International Journal of Social Science*, 6(2), 1169–1185.
- Kukul, V., Gokcearslan, S., & Gunbatar, M. S. (2017). Computer programming self-efficacy scale (CPSES) for secondary school students: Development, validation and reliability. *Educational Technology Theory and Practice*, 7(1), 158–179. doi:10.17943/etku.288493
- Lau, W. W. F., & Yuen, A. H. K. (2011). Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education*, 57(1), 1202–1213. doi:10.1016/j.compedu.2011.01.002

- Lee, Y.-J. (2011). Empowering teachers to create educational software: A constructivist approach utilizing Etoys, pair programming and cognitive apprenticeship. *Computers & Education*, 56(2), 527–538. doi:10.1016/j.compedu.2010.09.018
- Liao, Y.-K. C., & Bright, G. W. (1991). Effects of computer programming on cognitive outcomes: A meta-analysis. *Journal of Educational Computing Research*, 7(3), 251–268. doi:10.2190/E53G-HH8K-AJRR-K69M
- Lye, S. Y., & 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. doi:10.1016/j.chb.2014.09.012
- Mazman, S. G., & Altun, A. (2013). The effect of introductory to programming course on programming self-efficacy of CEIT students. *Journal of Instructional Technologies & Teacher Education*, 2(3), 24–29.
- Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (Moti). (2013). Learning computer science concepts with Scratch. *Computer Science Education*, 23(3), 239–264. doi:10.1080/08993408.2013.832022
- Monroy-Hernández, A., & Resnick, M. (2008). FEATURE Empowering kids to create and share programmable media. *Interactions*, 15(2), 50. doi:10.1145/1340961.1340974
- Pausch, R., Dann, W., & Cooper, S. (2000). Alice: A 3-D tool for introductory programming concepts. *Journal of Computing Sciences in Colleges*, 15(5), 107–116.
- Resnick, M. (2002). Rethinking learning in the digital age. In G. Kirkman (Ed.), *The global information technology report: Readiness for the networked world* (pp. 32–37). NY: Oxford University Press. doi:10.1353/cj.2007.0001
- Rubio, M. A., Romero-Zaliz, R., Mañoso, C., & de Madrid, A. P. (2015). Closing the gender gap in an introductory programming course. *Computers & Education*, 82, 409–420. doi:10.1016/j.compedu.2014.12.003
- Sáez-López, J. M., Román-González, M., & Vázquez-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. doi:10.1016/j.compedu.2016.03.003
- Scratch. (2018). About Scratch. Retrieved from <https://scratch.mit.edu/about>.
- Su, A. Y. S., Yang, S. J. H., Hwang, W. Y., Huang, C. S. J., & Tern, M. Y. (2014). Investigating the role of computer-supported annotation in problem-solving-based teaching: An empirical study of a Scratch programming pedagogy. *British Journal of Educational Technology*, 45(4), 647–665. doi:10.1111/bjet.12058
- Yukselturk, E., & Altiok, S. (2017). An investigation of the effects of programming with Scratch on the preservice IT teachers' self-efficacy perceptions and attitudes towards computer programming. *British Journal of Educational Technology*, 48(3), 789–801. doi:10.1111/bjet.12453
- Zuckerman, O., Blau, I., & Monroy-Hernández, A. (2009). Children's participation patterns in online communities: An analysis of Israeli learners in the Scratch online community. *Interdisciplinary Journal of E-Learning and Learning Objects*, 5, 263–274.