Computational Thinking Skills, Programming Self-Efficacies and Programming Attitudes of the Students

Mustafa Serkan Günbatar[1]

[1] Van Yüzüncü Yıl University

Abstract

The purpose of this research is to examine Computational Thinking (CT) skills, Computer Programming Self-Efficacy (CPSE), and Computer Programming Attitude (CPA) of middle school students who took the Information Technologies & Software (IT&S) courses and those who did not, and make various analyses according to the relationships between these variables. It is a correlational study and participants are 5th, 6th, 7th and 8th-grade middle school students. Computational Thinking Levels Scale, Computer Programming Self-Efficacy Scale, and Educational Computer Games Assisted Learning Coding Attitude Scale were used as data collection tools. Multivariable regression analysis and Multivariate Analysis of Variance were performed as statistical technics. As a result of the analysis, it was found that CPA and CPSE variables are significant predictors for CT skills. Both students who took IT&S course and those who did not take have moderate CPSE levels. But students who took IT&S course have a statistically significantly higher CPSE scores. In addition, it was observed that both those who took IT&S course and those who did not have moderate CPA and did not differ statistically, and students who took IT&S course had a high level of CT, while those who did not take the course were moderate and statistically different. It was also found that students who took IT&S courses made cumulative progress in terms of CPSE, CPA and CT variables compared to those who did not. The obtained results are discussed within the framework of middle school IT & S course curriculum held in Turkey in 2018.

Keywords: computational thinking skills; computer programming self-efficacy; computer programming attitude; middle school students; information technologies & software course

1. Introduction

Almost all of today's middle school students are growing in a world surrounded by digital equipment. Even if they do not have these digital equipment, they have information about what they do and how they are used. One of the theories that can best explain this is the Social Learning Theory by Albert Bandura (1971). Bandura mentions learning through modelling in social environment. Thanks to the informative function of the model, the learning product emerges and the learners learn by observing. The learner first perceives and interprets the situations in the social environment; then he remembers that they interpret and code into their long-term memory; then he practices his observations; finally, if he has the motivation to repeat this observation, he repeats the behavior at different times (Bandura, 1971). We can call this generation a digital native, born into the digital world and speaking the language of the digital world as its mother tongue (Prensky, 2001a). Communicating with digital natives who learn to use the technological language as their mother tongue in a social environment can be through well-designed games or activities in teaching environments (Prensky, 2001b). At this point, teaching by activities such as problem solving in the process comes to mind.

P21 (Partnership for 21st century learning), which has international members and aims to develop educational policies, has provided a framework for 21st century learners. The framework covers elements under two main themes: Learning outcomes and Support systems (P21, 2020). These elements, which are essential for all students at the point of being ready for the 21st century, are directly or indirectly related to technology. In parallel, institutions such as ISTE (International Society for Technology in Education) and CSTA (Computer Science Teachers Association) also provided an opinion on the concept of CT brought up by Wing (2006) and how to develop it at the K-12 level (ISTE, 2018; ISTE & CSTA, 2011). Turkish MoE (Ministry of Education), which was not indifferent to such studies, updated its middle school IT&S course curriculum in 2018. The feature of this new program is that it gives a large space to the problem solving and programming unit. When looking at their specific goals, it is seen that great importance is given to problem solving and CT skills (MoE, 2018a; MoE, 2018b).

In the literature, there are studies examining the relationship between CT and programming performance according to various variables (Shute, Sun & Asbell-Clarke, 2017; Oluk & Korkmaz, 2016; Durak & Saritepeci, 2018). These studies generally deal with variables related to cognitive structure. However, affective variables are also important for CT skills (Román-González, Pérez-González & Jiménez-Fernández, 2017; Román-González, Pérez-González, Moreno-León, & Robles, 2018) and it is recommended to establish regression models in the future studies to predict CT skills (Avcu & Ayverdi, 2020). In the literature review, the first study that comes to mind about the CT skills at the middle school level, CPSE (Computer Programming Self-Efficacy) and CPA (Computer Programming Attitude) were not studied in the same experiment together. From this point of view, the CPA, CPSE and CT of middle school students who took the IT&S course and those who did not take were examined, and various analyses were made according to the relationships between these variables. In this context, research questions of the study can be listed as follows.

RQ1: Are CPA and CPSE variables significant predictors for CT skills?

RQ2: What is the distribution of CT skills, CPA and CPSE measurements of students who took IT&S course and those who did not?

RQ3: Do the students who took the IT&S course and those who did not take differ according to the common effect of the CPA, CPSE and CT skills variables?

2. The study

2.1. Research Design

In this study, CT skills, CPSE and CPA of middle school students were measured. Then, the relationships between these variables were examined and it was determined whether the aforementioned measurements differ according to whether students took IT&S course or not. Therefore, it is a correlational study. As in this study, studies to reveal the relationships between variables are evaluated within the scope of the relational screening model (Fraenkel & Wallen, 2009).

2.2. Participants

Participants are the students who were studying in 5, 6, 7 and 8th-grade middle school in 2018-2019 spring semester in Van province of Turkey. Van is in the east of Turkey. The data were collected in the first quarter of 2019. Students were from four different middle public schools. Analysis was carried out with the data of 506 students in total, 221 of them are female and 285 of them are male. Students took IT&S course according to the infrastructure of computer laboratory in their schools.

2.3. Instrumentation

2.3.1. Computational Thinking Levels Scale

Computational Thinking Levels Scale (CTLS) was used to measure students' CT skills. The scale is a five-point Likert type and adapted by Korkmaz, Çakır & Özden (2016) for middle school students. The scale has five sub-dimensions. The Cronbach Alpha coefficient of the Creativity sub-dimension consisting of four items is .640; the four-point Algorithmic Thinking sub-dimension was .762; the four-point Collaboration sub-dimension .811;

the four-point Critical Thinking sub-dimension was .714; the six-item Problem Solving sub-dimension was calculated as .867. The Cronbach Alpha coefficient of the 22-item scale is .809. As a result of Confirmatory Factor Analysis performed in the adaptation process, goodness of fit values were determined as RMSEA = .074, S-RMR = .078, GFI = .89, AGFI = .84, CFI = .91, NNFI = .91, IFI = .90. According to these results, it has been stated by the researchers that the scale model has acceptable fit values. In addition, according to the results, it was found that the scale is a reliable and valid measurement tool to measure Turkish students' CT skills.

2.3.2. Computer Programming Self-Efficacy Scale

Computer Programming Self-Efficacy Scale (CPSES) developed by Kukul, Gökçearslan & Günbatar (2017) was aimed to measure students' CPSE. As a result of the exploratory factor analysis, the scale took its final shape with 31 items and consists of one factor. Cronbach Alpha coefficient of the five-point likert type scale is .950. As a result of confirmatory factor analysis Goodness of fit values were found as: $X^2 / df = 1.84$; RMSEA = .06; NFI = .95; NNFI = .98; RMR = .068; CFI = .98; IFI = .98; GFI = .82 and AGFI = .79. According to the obtained data model for scale, it was found that Turkish middle school students have compliance with acceptable values and the model has a high level of reliability.

2.3.3. Computer Programming Attitude Scale

Educational Computer Games Assisted Learning Coding Attitude Scale was used to determine the students' CPAs. The Scale was developed by Keçeci, Alan & Kırbağ-Zengin (2016). The scale, consisting of a total of 28 items, is a five-point Likert type. To determine the sub-factor structure of the scale, exploratory factor analysis was performed. The scale consists of three sub-factors. The Cronbach Alpha coefficient of the 12-item sub-factor, which is called as a request against coding learning, is .840; the 11-item sub-factor, which is called the interest for the use of computer games in education for education purposes is .736; the five-item sub-factor called concern about the asocialization of the computer is .481. The Cronbach Alpha coefficient of the whole scale is .833. As a result of the analysis, it has been revealed that the scale is sufficiently valid and reliable.

2.4. Data Analysis

Firstly, descriptive statistics related to CT, CPSE and CPA measurements of the students were presented. Then, multivariable regression analysis was performed to determine significant predictors for CT. Multivariate Analysis of Variance (MANOVA) was performed to determine if students who took IT&S courses and those who did not take differed according to the common effect of CT, CPSE and CPA measurements. Finally, One-way analysis of Variance (ANOVA) was performed to compare the CTS, CPSE and CPA measurements of students who took IT&S courses and those who did not.

3. Findings

RQ1: Are CPA and CPSE variables significant predictors for CT skills?

Table 1. Result of multiple regression analysis conducted for CT skills predict

Variable	B	Standard Error	β	t	p	Zero-order r	Partial r
Fixed Value	1.188	.166		7.169	.000		
CPSE	.465	.037	.485	12.668	.000	.530	.492
CPA	.210	.046	.175	4.578	.000	.300	.200
R=.557		R²=.310					
F$_{(2,503)}$=113.026		p=.000					

For the reference points, Pearson correlation coeffients (r) can take values from -1 to +1. We can interpret values .10 to .29 as small (low), .30 to .49 as medium (moderate), .50 to 1.0 as large (high) (Pallant, 2007). According to Table 1, it is seen that there is a positive and moderate (r = .530) correlation between CT and CPSE according to bilateral and partial correlations between predictor variables and predicted variable, and when CPA variable is controlled, partial correlation between CT skill and CPSE is similarly positive and moderate (r = .492).

It is seen that there is a positive and moderate (r = .300) relationship between CT skill and CPA, and when the CPSE variable is controlled, there is a positive and low (r = .200) relationship between CT skill and CPA.

When CPSE and CPA variables are considered together, they offer a moderate and significant relationship with CT skills scores, R = .557, R^2 = .310, p <.01. The CPSE and CPA variables explain approximately 31% of the total variance in the CT skills.

According to the standardized regression coefficient (β), the significance level of the CPSE variable for CT skills is higher than for CPA. When the t-test results related to the significance of the regression coefficient are examined, it is seen that both variables are significant predictors of CT skills (p < .01). Regression equation for predicting CT skills is given below.

$$CT = 1.188 + .465 * CPSE + .210 * CPA \tag{1}$$

RQ2: What is the distribution of CT skills, CPA and CPSE measurements of students who took IT&S course and those who did not?

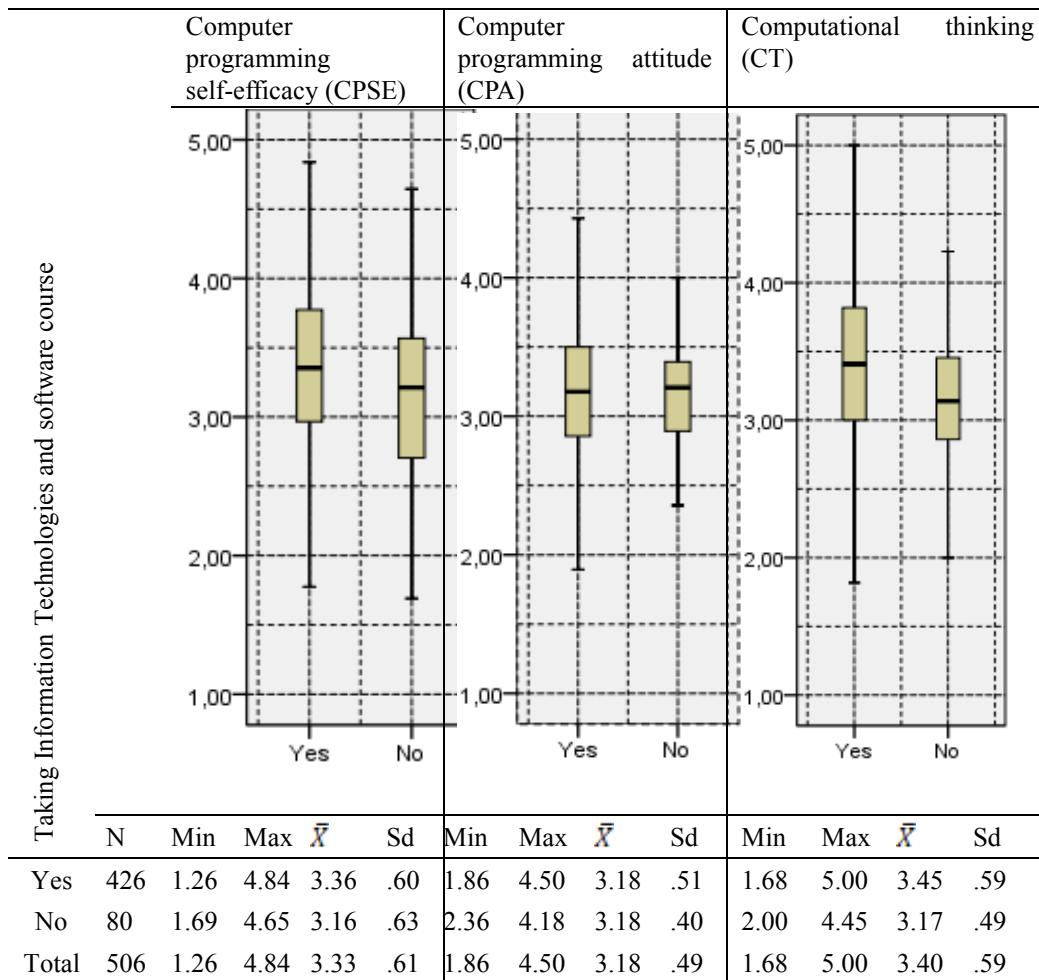|	N	Min	Max	$\bar{X}$	Sd	Min	Max	$\bar{X}$	Sd	Min	Max	$\bar{X}$	Sd
Yes	426	1.26	4.84	3.36	.60	1.86	4.50	3.18	.51	1.68	5.00	3.45	.59
No	80	1.69	4.65	3.16	.63	2.36	4.18	3.18	.40	2.00	4.45	3.17	.49
Total	506	1.26	4.84	3.33	.61	1.86	4.50	3.18	.49	1.68	5.00	3.40	.59

Figure 1. Distribution of CPSE, CPA and CT measurements of students who took IT&S courses and those who did not

In figure 1, there are values related to the distribution of CPSE, CPA and CT skills measurements of students who took IT&S course and those who did not take. 426 of the students participating in the study took IT&S course and 80 of them did not. Mean ranges from five-point likert scales can be interpreted for 1.00-1.79 as *very low*; 1.80-2.59 as *low*; 2.60-3.39 as *moderate*; 3.40-4.19 as *high*; 4.20-5.00 as *very high*. While the average CPSE of students who took the course is $\bar{X}$ = 3.36 (moderate), the average of those who did not take the course is $\bar{X}$ = 3.16 (moderate). CPA averages of students who took the course and those who did not are very similar and their average is $\bar{X}$ = 3.18 (moderate). While the average of the CT skills of the students who took the course is $\bar{X}$ = 3.45 (high), the average of those who did not take is $\bar{X}$ = 3.17 (moderate).

RQ3: Do the students who took the IT&S course and those who did not take differ according to the common effect of the CPA, CPSE and CT skill variables?

It was tested whether the data set belonging to the students, who took IT&S course and those who did not, provided the assumptions of the MANOVA analysis. It was seen that assumptions were provided. As a result of the analysis, according to the common effect of the dependent variables a significant difference was found among the students in terms of whether they took IT&S courses (F $_{(3,502)}$ = 6.071, p = .000; Wilks' Lambda (^) = .965; Partial eta squared = .035).

Table 2. Mean, Standard Deviation Values and ANOVA Results of CT, CPSE and CPA Measurements according to taking IT&S course

	IT&S course	N	$\overline{x}$	Sd	df	F	p	Partial Eta Squared
CT	Yes	426	3.4481	.59261	1-504	15.737	.000*	.030
	No	80	3.1687	.49326				
CPSE	Yes	426	3.3645	.60484	1-504	7.890	.005*	.015
	No	80	3.1564	.62530				
CPA	Yes	426	3.1799	.50704	1-504	.002	.965	.000
	No	80	3.1773	.39746				

*p< 0.01

Table 2 shows the ANOVA results in which dependent variables are compared according to whether students took IT&S course or not. CT and CPSE variables of the students who took IT&S course have significantly higher levels than those who did not (p < .01). In terms of CPA, students who took the IT&S course and those who did not take do not differ (p> .05).

4. Discussion of Findings

This research was carried out to examine CT, CPSE, and CPA of middle school students who took IT&S course and those who did not take and to make various analyzes according to the relationships between these variables. The results obtained revealed that the variables CPA and CPSE are significant predictors for CT skills. In terms of CT skills predict, the CPSE variable is stronger than the CPA variable. CT, which is derived from computer science (Yadav, Stephenson & Hong, 2017) and identified with programming, has no clear definition (Computing at school, 2020). However, it is known that CT is not the same as programming skills. It can be said that making a computer program creates an infrastructure to think computationally (Shute, Sun & Asbell-Clarke, 2017). In parallel, There are studies that show results regarding the relationship between Programming performance and CT such as; students who use block-based programming tools develop their CT skills as much as their programming skills improve (Oluk & Korkmaz, 2016); the variable of academic success of Information technologies can significantly predict CT skills (Durak & Saritepeci, 2018). These variables, which are related to programming performance, can be described as cognitive factors. For CT skills, non-cognitive factors such as self-efficacy and attitude are determinative as well as cognitive factors (Román-González et al., 2018). In the literature, there are studies on non-cognitive factors that support the results of this study. It was found that there is a moderate significant relationship between CPA and CPSE (Çoban, Korkmaz, Çakır & Erdoğmuş, 2020); and a moderate significant relationship between CT and CPSE (Durak, Yılmaz, & Yılmaz, 2019) and CPSE predicted CT significantly (Avcu & Ayverdi, 2020). The importance of the data obtained within the scope of this study is that the CPSE and CPA variables, which come to mind as the non-cognitive factor when it comes to CT, are considered together for the middle school students, and this study reveals the order of their importance.

When CPSE levels were analyzed, it was seen that both students who took IT&S course and those who did not take had moderate self-efficacy. Although both groups seem to be at the same level, comparisons in terms of their averages showed that the students who took the course had higher self-efficacy statistically significantly. The concept of self-efficacy can be explained as "the individual's perception of the capacity to perform the required effort to perform a particular task" (Bandura, 1997) and there is a mutual relationship with the performance variable (Lishinski, Yadav, Good & Enbody, 2016). Prior programming experience is an important variable for CPSE (Mazman & Altun, 2013; Kittur, 2020). Effect of programming experience on self-efficacy was revealed, after four weeks of STEM-supported training (Feldhausen, Weese, & Bean, 2018) or for a period

of algorithm and coding training (Mıhçı-Türker & Pala, 2020; Okal, Yıldırım, & Timur, 2020). Considering the situation that people think that computer programming is complex and difficult (Yükseltürk & Altıok, 2017), it is an expected result that students who did not take IT&S course before and therefore did never perform computer-programming performance would have lower CPSE perceptions.

When the CPAs of the students were examined, it was seen that both the students who took IT&S course and those who did not had very close averages. Both groups have moderate CPA. There is no statistically significant difference according to the comparisons over their averages. In the literature, there are studies that demonstrate that students' programming attitudes have increased as a result of the programming instruction carried out with computer programming tools such as mBlock and Scratch working with drag and drop logic (Günbatar & Karalar, 2018; Yükseltürk & Altıok, 2017). In addition, there are studies revealing that tools such as scratch used in programming instruction do not provide similar attitude development with the traditional method (Çetin, 2016) and that the attitudes of both gifted and normal students towards computer-aided coding are similar (Toklu, 2019). Based on these results, it is thought that CPA can develop at the end of the instruction, which is carried out either with traditional method or with various programming tools. However, it has been demonstrated that people who are interested in computer programming for many years do not differ from those who are new in this business in terms of attitude (Günbatar, 2018). Attitude is the mental state in which the individual pre-chooses the behaviour in the face of certain situations (Gagne, 1985). Attitudes guide human behaviour, and they constitute an important place within the scope of affective domain learning (Balaban-Salı, 2006). There are several levels of affective domain learning, ranging from a basic level, such as having knowledge about a particular situation, to a complex situation ranging from creating a life philosophy (Krathwohl, Bloom & Masia, 1964). Getting information about the attitude of an individual about a certain subject directly reveals his attitude. However, it is not possible for everyone to present objective information, and different approaches to obtaining information (such as collecting information from others, reviewing records) can also be used (Balaban-Salı, 2006). In the literature, it is mentioned that there is a development in programming attitude after basic programming training, and there is no difference between those who are interested in programming for many years and those who are new to this business in terms of programming attitude. However, the number of studies on computer programming and student attitudes is not high (Çetin & Ozden, 2015). Considering the complex nature of the attitude, it is recommended to carry out new studies (using different approaches to obtain information if possible).

When CT measurements are examined, students who took IT&S course had high level of CT skills; those who did not take the course were found to be at a moderate level. Comparison results based on averages also support this level differentiation. Students who took the course have significantly higher CT skills than those who did not take the course. In the process of collecting data within the scope of the study, MoE Board of Education Discipline has recently organized the IT&S course curriculum at middle school level. Therefore, the data were collected during the first applications of the renewed curriculum. Within the scope of the specific objectives of the renewed curriculum, main expectations are: The students' acquisition and development of problem solving and computational thinking skills; To be able to express their verbal and visual expressions by developing an understanding of algorithm design; To choose the appropriate programming approach to solve problems and apply them; Creating technical knowledge in programming; Ability to use at least one of the programming languages; Developing innovative and original projects for the solution of the problems encountered in daily life (problems faced by elderly and disabled people, etc.) ". In this context, a "Problem solving and programming" unit covers a total of 72 hours, covering 50% for the 5th and 6th grades and 44% for the 7th and 8th grades. It is planned to achieve the above-mentioned special objectives within the scope of this unit (MoE, 2018a; MoE, 2018b). The CT skills difference between the students who took IT&S course and those who did not take the course, and the problem solving and programming unit, which is given an important place within the scope of the curriculum, are an indication that the new curriculum can achieve its purpose.

Students who took the course and those who did not differ according to the common effect of CPSE, CPA and CT skills. In other words, the students who took this course make progress cumulatively in terms of these three variables compared to those who did not. As a result of their study, Román-González, Pérez-González, Moreno-León, & Robles (2018) emphasized non-cognitive factors related to CT. They provided suggestions for training policies for the development of CT skills. Accordingly, besides cognitive psychological construct such as problem solving skill, non-cognitive factors such as self-efficacy and personality should also be considered. Within the scope of this study, this differentiation obtained with the common effect of self-efficacy, attitude and CT skills confirms the idea of Román-González, Pérez-González, Moreno-León, & Robles (2018). Turkey,

revealed the importance of the CPSE and CPA variables as well as CT skills to achieve the specific objectives within the scope of the problem-solving and programming unit with its renewed IT & S course curriculum. Based on these results, especially the importance of problem-solving and programming unit's objectives has emerged. Updated middle school IT&S course curriculum requirements must be accomplished by all institutions in Turkey. Curriculum developers in other countries must recognize the importance of the problem-solving and programming subjects. They should make decisions to equip students with the skills with this context.

References

Avcu, Y.E. & Ayverdi, L. (2020). Examination of the computer programming self-efficacy's prediction towards the computational thinking skills of the gifted and talented students. *International Journal of Educational Methodology, 6*(2). 259-270. doi: 10.12973/ijem.6.2.259

Balaban-Salı, J. (2006). Tutumların öğretimi. In A. Şimşek (Ed.), *İçerik Türlerine Dayalı Öğretim* (pp 133-162). Ankara: Nobel Yayın Dağıtım.

Bandura, A. (1971). *Social learning theory*. New York: General Learning Press.

Bandura, A. (1997). *Self-efficacy: The exercise of control*. New York: W. H. Freeman.

Çetin, I. (2016). Preservice teachers' introduction to computing: exploring utilization of scratch. *Journal of Educational Computing Research, 54*(7), 997-1021. doi: 10.1177/0735633116642774

Çetin, I., & Özden, M. Y. (2015). Development of computer programming attitude scale for university students. *Computer Applications in Engineering Education*, *23*(5), 667-672.

Computing at school (2020). *Computational thinking*. Retrieved July 9, 2020 from https://community.computingatschool.org.uk/files/8261/original.pdf

Çoban, E., Korkmaz, Ö., Çakır, R., & Erdoğmuş, F. U. (2020). Attitudes of IT teacher candidates towards computer programming and their self-efficacy and opinions regarding to block-based programming. *Education and Information Technologies*, *25*(5), 4097-4114. doi: 10.1007/s10639-020-10164-w

Durak, H. Y., & Saritepeci, M. (2018). Analysis of the relation between computational thinking skills and various variables with the structural equation model. *Computers & Education*, *116*, 191-202. doi: 10.1016/j.compedu.2017.09.004

Durak, H. Y., Yılmaz, F. G. K., & Yılmaz, R. (2019). Computational thinking, programming self-efficacy, problem solving and experiences in the programming process conducted with robotic activities. *Contemporary Educational Technology*, *10*(2), 173-197. doi: 10.30935/cet.554493

Feldhausen, R., Weese, J. L., & Bean, N. H. (2018, February). Increasing student self-efficacy in computational thinking via STEM outreach programs. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education,* Baltimore Maryland, USA.

Freankel, J.R. & Wallen, N. E. (2009). *How to design and evaluate research in education* (7th edition). New York: McGraw-Hill.

Gagne, R. M. (1985). *The conditions of Learning and theory of instruction* (4th ed.). New York, NY: CBS College Publishing.

Günbatar, M. S. (2018). Examination of undergraduate and associate degree students' computer programming attitude and self-efficacy according to thinking style, gender and experience. *Contemporary Educational Technology*, *9*(4), 354-373. doi: 10.30935/cet.471004

Günbatar, M. S., & Karalar, H. (2018). Gender differences in middle school students' attitudes and self-efficacy perceptions towards mBlock programming. *European Journal of Educational Research*, *7*(4), 925-933. doi: 10.12973/eu-jer.7.4.923

ISTE (2018). *ISTE standards for students*. Retrieved April 11, 2018 from https://www.iste.org/standards/for-students

ISTE and CSTA (2011). *Operational definition of computational thinking for K-12 thinking*

operational-definition-flyer. Retrieved July 16, 2020 from http://www.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf?sfvrsn¼2

Keçeci, G., Alan, B., & Kırbağ-Zengin, F. (2016). Educational computer games assisted learning coding attitude scale: validity and reliability study. *Education Sciences*, *11*(3), 184-194. doi: 10.12739/NWSA.2016.11.3.1C0661

Kittur, J. (2020). *Measuring the programming self-efficacy of Electrical and Electronics Engineering students*. Retrieved July 27, 2020 from https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9031745

Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2016). Computational thinking levels scale (CTLS) adaptation for secondary school level. *Gazi Journal of Educational Science*, *1*(2).

Krathwohl, D. R., Bloom, B. S., & Masia, B. B. (1964). *Taxonomy of Educational Objectives, Handbook II: Affective Domain.* New York: Longman.

Kukul, V., Gökçearslan, Ş., & Günbatar, 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

Lishinski, A., Yadav, A., Good, J., & Enbody, R. (2016, August). Learning to program: Gender differences and interactive effects of students' motivation, goals, and self-efficacy on performance. In *Proceedings of the 12th Annual International ACM Conference on International Computing Education Research,* Melbourne VIC, Australia.

Mazman, S. G., & Altun, A. (2013). Programlama–I Dersinin BÖTE Bölümü Öğrencilerinin Programlamaya İlişkin Öz Yeterlilik Algıları Üzerine Etkisi. *Journal of Instructional Technologies & Teacher Education*, *2*(3), 24-29.

MoE (2018a). *Bilişim teknolojileri ve yazılım dersi öğretim programı (5. ve 6. Sınıflar).* Retrieved June 25, 2020 from http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bilişim Teknolojileri ve Yazılım 5-6. Sınıflar.pdf

MoE (2018b). *Bilişim teknolojileri ve yazılım dersi öğretim programı (7. ve 8. Sınıflar).* Retrieved June 25, 2020 from http://mufredat.meb.gov.tr/Dosyalar/2018813171426130-2-2018-81Bilişim Teknolojileri ve Yazılım Dersi (7 ve 8. Sınıflar).pdf

Mıhçı-Türker, P. & Pala, F.K. (2020). The effect of algorithm education on students' computer programming self-efficacy perceptions and computational thinking skills. *International Journal of Computer Science Education in Schools. 3*(3). 1-11. doi: 10.21585/ijcses.v3i3.69

Okal, G., Yıldırım, B. & Timur, S. (2020). The effect of coding education on 5th, 6th and 7th grade students' programming self-efficacy and attitudes about technology. *Educational Policy Analysis and Strategic Research, 15*(2), 143-165. doi: 10.29329/epasr.2020.251.8

Oluk, A. & Korkmaz, Ö. (2016). Comparing students' scratch skills with their computational thinking skills in terms of different variables. *International Journal of Modern Education and Computer Science. 11.* 1-7. doi: 10.5815/ijmecs.2016.11.01

P21 (2020). *P21 Frameworks & Resources*. Retrieved February 28, 2020 from http://www.battelleforkids.org/networks/p21/frameworks-resources

Pallant, J. F. (2007). *SPSS survival manual: A step-by-step guide to data analysis with SPSS* (3rd edition). New York: Open University Press.

Prensky, M. (2001a). Digital natives, digital immigrants. *On The Horizon*, *9*(5).

Prensky, M. (2001b). Digital natives, digital immigrants part 2: Do they really think differently?. *On The Horizon, 9*(6).

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

Román-González, M., Pérez-González, J. C., Moreno-León, J., & Robles, G. (2018). Extending the nomological network of computational thinking with non-cognitive factors. *Computers in Human Behavior*, *80*, 441-459. doi: 10.1016/j.chb.2017.09.030

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

Toklu, E. (2019). *Game Desing and Coding Training For 7-11 Year Old Gifted Students – Kodu Game Lab Example* (Unpublished master's thesis). Uludağ University, Bursa-Turkey.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*, 33-35.

Yadav, A., Stephenson, C. & Hong, H. (2017). Computational thinking for teacher education. *Communications of the ACM*, *60,*(4), 55-62. doi: 10.1145/2994591

Yükseltürk, E., & Altıok, 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