Informatics in Education, 2019, Vol. 18, No. 2, 281–296

DOI: 10.15388/infedu.2019.13

Factors Affecting Students' Attitudes toward Computer Programming

Melih Derya GURER, Ibrahim CETIN, Ercan TOP
Facuty of Education, Bolu Abant Izzet Baysal University, Turkey
e-mail: mdgurer@gmail.com, ibretin@hotmail.com, ercantop@gmail.com

Received: February 2019

Abstract. The aim of this study was to investigate the factors affecting the pre-service computer science teachers' attitudes towards computer programming (ATCP). The sample consists of 119 pre-service teachers at a public state university. The influences of students' demographic characteristics (gender, grade level, and high school type), their achievement in computer programming courses, perceived learning, and computer programming self-efficacy on their ATCP were tested using multiple linear regression. Descriptive, correlation and regression analyses revealed three findings: (1) students had moderately high ATCP, (2) their ATCP had significant correlations with their achievement in computer programming courses, computer programming self-efficacy, and perceived learning, and (3) three variables (achievement in computer programming courses, computer programming self-efficacy, and perceived learning) were significant predictors of their ATCP.

Keywords: computer programming attitude, self-efficacy, achievement, perceived learning.

1. Introduction

Recently, two main trends relating to computational thinking have emerged in the field of education; one at K-12 level, information and communications technology (ICT) education, and the other in academic community. Students are expected to be knowledgeable about ICT to survive in the twenty-first century (Norris, 2001; Trilling & Fadel, 2009), and consequently, considerable changes in ICT related courses even at K-12 level have been observed. In addition to holding the core background and skills into ICT tools, students are expected to construct ICT tools themselves by applying the basic notions of computer science (Hubwieser, Armoni, Giannakos, & Mittermeir, 2014). Considering this demand, computing courses have become compulsory for students from ages 5–16 in the UK (Department for Education, 2014). Likewise, in the USA, computer science standards have been in use since they were published by the Computer Science Teacher Association in 2011 (Seehorn *et al.*, 2011). What is more, 12 European countries are already offering computing related courses in their curriculum and six more are planning to improve their curriculum with respect to computer science courses.

Since Wing's (2006) influential paper on computational thinking, scholars have focused on two main questions including what computational thinking is and how educators can help their students improve their computational thinking skills. Cuny, Snider, and Wing (2010) define computational thinking as "the thought process involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent" (as cited in Wing, 2011, p. 20), while Selby and Woollard (2010) consider computational thinking as a thought process composed of five main concepts: abstraction, decomposition, algorithmic thinking, evaluation and generalization. To some, students do not have to be directly involved in computer programming to be able to develop concepts of computational thinking, as suggested in the Computer Science Unplugged movement which considers the ways to help students improve their computational thinking abilities without having to use computers (Bell, Rosamond, & Casey, 2012). On the other hand, some other researchers in the field claim computer programming is one of the fundamental tools to improve computational thinking (Armoni, Meerbaum-Salant, & Ben-Ari, 2015). Also, as Brennan and Resnick (2012) proposed, concepts used for programming could be regarded as the basic dimensions of computational thinking.

Such demands for computing courses and interest in computational thinking have led researchers to seek answers for one fundamental question: How can students be introduced to computer programming? (Cetin & Ozden, 2015). As far as can be predicted, computer programming is a complex skill that grows only gradually, and it presents a set of challenges in five main areas (Boulay, 1986): general orientation to programming or what programming is about; notional machine – an abstraction of a computer by learners; syntax and semantics of programming; standard solutions to the standard problems of computer programming – e.g. algorithms for searching and sorting; and finally, pragmatics of programming – designing, developing, testing, and debugging a program. Novice learners need to consider all these areas simultaneously while learning about and practicing programming, which brings about a significant amount of cognitive load (Boulay, 1986).

Studies on students' learning process have been conducted in an effort to help them improve their programming understanding utilizing different approaches, e.g. novice-expert comparisons (Wiedenbeck, Fix, & Scholtz, 1993), students' misconceptions (Ragonis & Ben-Ari, 2005), students' understanding (Götschi, Sanders, & Galphin, 2003), and meta-cognition (Cetin, Sendurur, & Sendurur, 2014). Nevertheless, students' attitudes towards computer programming (ATCP) have not received much attention in the field (Cetin & Ozden, 2015). One reason for this could be associated with the difficulties encountered while assessing attitude, such as defining the construct or developing assessment tools. In response to such need, Cetin and Ozden (2015), holding a tripartite view regarding attitude including the dimensions of cognition, affection, and behavior, constructed a computer programming attitude scale.

Although under-explored in the field, confounding findings as to the potential relationship between students' ATCP and their achievement in programming exist. There are some correlational studies revealing a positive relationship between students' achievement and their ATCP (Baser, 2013a; Ma & Kishor, 1997), while some experimental studies investigating the effect of an intervention on students' achievement in programming and their ATCP do not suggest a straightforward relation as in Cetin and Andrews-Larson's study (2016) in which an intervention based on visualization construction had a significant effect on students' achievement but did not result in a significant increase in their ATCP, further confirmed in Cetin's (2016) and Korkmaz's (2016) studies. Thus, the potential link between students' ATCP and their performance in computer programming needs to be further investigated.

Gender, as one of the potential factors that might have an effect on students' ATCP, need to be explored within this framework as well, because various studies reveal women are underrepresented in computing related jobs and computer science majors (e.g. Cukier, Shortt, & Devine, 2002; Doube & Lang, 2012; Singh, Allen, Scheckler, & Darlington, 2007), which might be resulting from some cultural and environmental conditions potentially influencing women's preferences and choices. Women's relatively less positive ATCP might be the reason for their under-representation though (Gurer & Camp, 2002; Stoilescu & Egodawatte, 2010). Baser (2013b) reports that males have more positive ATCP than females; Ozyurt and Ozyurt (2015), and Korkmaz and Altun (2013) present similar results. Despite such studies pointing out gender differences in ICT areas, it has also been shown that gender differences in ICT use have diminished in general (Alsadoon, 2013; Huang, Hood, & Yoo 2013; Top, Yukselturk, & Cakir, 2011).

Self-efficacy is a potential area in exploring factors affecting students' ATCP. Bandura (1977) defines perceived self-efficacy as the belief in one's capabilities to achieve in specific situations or accomplish a task, and states that efficacy beliefs "... influence how people feel, think, motivate themselves, and act" (p.116). As such, in theory, self-efficacy beliefs have the potential to influence attitudes as attitudes comprise affect, cognition, and behavior (Aiken, 2002); however, this potential relationship has not been investigated thoroughly in the field of computer programming education (Askar & Davenport, 2009). Studies to date suggest that university students' self-efficacy regarding computer use is positively associated with their attitudes towards computers (Kinzie, Delcourt, & Powers, 1994) and that students' attitudes towards the internet and their internet self-efficacy are correlated (Wu & Tsai, 2006). Likewise, it has been shown that the success of students in programming lessons depends on their ATCP and computer programming self-efficacy (Yagci, 2016), and there is a positive correlation between students' ATCP and their computer programming self-efficacy (Ozyurt & Ozyurt, 2015). Further studies need to consider this relationship in explaining students' computer programming abilities.

Students' educational background can be another area in exploring factors affecting their ATCP. Ozyurt and Ozyurt (2015) report that university students' ATCP showed significant differences regarding their grade level such that the sophomores showed more positive ATCP than the freshmen. Their majors might as well be influential as shown in Korkmaz and Altun's study (2013), which reveals that of the students from three different departments including computer engineering, electrical and electronic engineering, and computer science and instructional technology, the computer engineering students had more positive ATCP than did the others.

One final area of potential relevance to students' ATCP could be their learning perception. Alavi (1994) defines learning perception as changes in learners' knowledge levels and skill perceptions before and after learning experiences. Views regarding the validity of such self-report as a measure of learning vary in the field and conflicts or conflictual arguments about actual performance and achievement versus learning perception exist. Although Koriat and Bjork (2005) and Winberg and Hedman (2008) do not see perceived learning as a good indicator of actual learning and assert that it may be different from actual achievement, Rovai, Wighting, Baker, and Grooms (2009) and Metcalfe (2009) state that learners can monitor their learning and therefore perceived learning can be a valid measure of achievement. In their line of reasoning, Rovai *et al.* (2009) rely on the idea that perceived learning has three components including cognitive, affective, and psychomotor learning, among which affective domain represents interests, opinions, emotions, attitudes, and values rather than mental abilities. Hence, considering its scope, students' perceived learning in a specific course is expected to influence their attitudes towards the subject, as well as in computer programming courses.

Although ATCP has attracted some interest, the research on the factors that shape the pre-service computer science teachers' ATCP is still limited. The current study examines the factors that correlate with students' ATCP, and how well gender, high school type, grade level, achievement in courses on computer programming, computer programming self-efficacy, and perceived learning predict students' ATCP.

As such, this study attempts to answer the following research questions:

1. What are the pre-service computer science teachers' ATCP, achievement in courses on computer programming, computer programming self-efficacy, and perceived learning?
2. To what extent does each predictor variable (gender, high school type, grade level, achievement in courses on computer programming, computer programming self-efficacy, and perceived learning) correlate with the students' ATCP?
3. To what extent do gender, high school type, grade level, achievement in courses on computer programming, computer programming self-efficacy, and perceived learning predict the students' ATCP and which variables are significant predictors of the students' ATCP?

2. Method

The purpose of the study is to reveal the extent of the selected variables (grade, school type, gender, achievement in computer programming lessons, computer programming self-efficacy, and perceived learning) account for students' ATCP. Hence, a correlational study design was adopted for this study.

2.1. *Participants*

The participants of this study were 119 pre-service teachers (Table 1) who were from the Computer Education and Instructional Tecnhnology department in a public state university in Turkey. During undergraduate education, they take 15 courses (out of 49) on information technology, computer programming, and instructional technology. After

Table 1

Demographic information about participants (n = 119)

Variable	Frequency	Percentage
Gender		
Male	59	49.6
Female	60	50.4
High School Type		
Vocational	84	70.6
General	35	29.4
Grade Level		
Sophomore	35	29.4
Junior	35	29.4
Senior	49	41.2

graduation, they can work as computer science teachers in secondary and high schools, instructional designers, and in jobs related with information and communication technology. Table 1 indicates the demographics of the participants. The students participated in the study were sophomore (N = 35), junior (N = 35), and senior (N = 49) students. Freshmen students were excluded as they don't have programming courses in their first year of schooling. Age of 97% of the students ranged from 19 to 24 and mean of their age was 21.80. The number of females (50.40%) was nearly the same with the number of male students (49.60%). However, most of the students (70.60%) were graduated from vocational high school in which computer programming courses were offered. Students graduated from vocational high schools mainly studied a four-year information technology program. During high school, they took courses about foundations of information and communication technology, software, and technical drawing. In addition to the common compulsory courses, they take some elective courses, such as, database management, networking, web design, computer programming, and system maintenance and repairment depending on their sub-program.

2.2. *Data Collection*

Data, except the academic achievement in computer programming lessons, were collected through online surveys at the end of the spring semester. Surveys were electronically sent to the individual students and only volunteers completed the surveys. The following online instruments were used to collect the relevant data; computer programming self-efficacy scale (CPSE), perceived learning scale (PL), and attitudes toward computer programming (ATCP) scale.

CPSE was originally developed by Ramalingam and Wiedenbeck (1998) to investigate computer programming self-efficacy scores of higher education students in C++ programming course. The reliability of the original scale was determined to be high (α = .98). The original scale had four factors; (1) independence and persistence, (2) complex programming tasks, (3) self-regulation, and (4) simple programming tasks and the reliability scores of each factor were changing between .86 and .94. Altun and Mazman (2012) adopted the original scale which resulted with nine items within two factors (ability to perform simple programming tasks and ability to perform complex programming tasks). The reliability of scale was calculated as .93 by Altun and Mazman (2012). In the current study, the internal consistency coefficient for this scale was calculated as .88.

In order to measure pre-service teachers' perceived learning in computer programming courses, PL was administered. Initially, PL was developed by Rovai *et al.* (2009) to reveal students' perceived learning in a course. The initial form was consisted of three dimensions; perceived cognitive, affective, and psychomotor learning. The reliability coefficient of the original scale was determined to be high ($\alpha = .79$). Top, Yukselturk and Inan (2010) adopted the original scale, and it was resulted in nine items with the reliability of .81. In the current study, the Cronbachs' alpha reliability coefficient of PL scale was found out as .70.

ATCP scale was originally developed by Cetin and Ozden (2015) to assess higher education students' attitudes toward computer programming. The 5-point Likert type scale had 18 items within three factors (affection, cognition, and behavior). Reliability scores of dimensions of ATCP scale were ranged between .80 and .90. The internal reliability of original scale was determined to be high, .94. In the current study, the overall Cronbach's alpha coefficient of ATCP scale was found out as 0.93.

Another datum for this study was students' academic achievement. Computer programming achievement scores of the students were calculated based on their last two computer programming courses' grades. Depending on their grade level, mean score of two of the following courses were used to calculate achievement scores of the students; Programming Language I, Programming Language II, and Internet Based Programming. The courses were presented to the students in their third, fourth and fifth semesters, respectively. In these courses, C, C++, and PHP were taught respectively. These courses were given in the Computer Education and Instructional Tecnhnology Department at a public state university. The main goals of the courses were helping pre-service teachers become knowledgeable and competent in the fundamental concepts and functions of computer programming.

2.3. *Data Analysis*

Before the data were subjected to statistical analysis, the negative items in the data collection instruments were reversed. The data gathered with online instruments and achievement in programming languages courses were analyzed initially by descriptive statistics such as mean and standard deviation. The normality of data was tested based on the assumption that in a normally distributed sample, 95% of z-scores of skewness and kurtosis should lie between −1,96 and +1,96 (Field, 2009). Although only the z-score of self-efficacy was higher than 1,96, the normality test of Kolmogorov-Smirnov indicated a normal distribution for self-efficacy. Hence, it was accepted that the data of each factor were normally distributed. Then, using the statistical analysis software SPSS 20.0, bivariate and multivariate analyses were conducted on the data. For example, Pearson Product-Moment Correlation was run to examine the correlation between the pre-service teachers' gender, grade, school type, achievement in programming languages courses, computer programming self-efficacy perceived learning, and ATCP. Additionally, multiple-regression analysis was run to assess how well the pre-service teachers' ATCP can be explained in terms of the selected variables (pre-service teachers' gender, grade level, high school type, achievement in programming languages courses, computer programming self-efficacy, and perceived learning).

3. Findings

3.1. *Descriptive Statistics*

Pre-service computer science teachers' mean scores on achievement in programming language courses, computer programming self-efficacy, perceived learning and ATCP are indicated in Table 2. Students attained moderately high scores on the computer programming self-efficacy ($\bar{X}$ = 3.87), perceived learning ($\bar{X}$ = 3.61) and ATCP ($\bar{X}$ = 3.73) scales on the 1–5 Likert measurement. Additionally, mean of the students' academic achievement scores is 67.41 over 100 points.

3.2. *Correlations among Variables*

In order to analyze the correlations between students' ATCP and the independent variables in the regression model, Pearson correlation coefficients were calculated. As seen in the Table 3, four variables were significantly correlated with students' ATCP. The pre-service computer science teachers' ATCP had statistically significant correlations with their achievement mean in computer programming courses (r(119) = .47), their computer programming self-efficacy (r(119) = .74), and perceived learning (r(119) = .71). On the other hand, the correlations between ATCP and grade level, between ATCP and high school type, and between ATCP and gender were not statistically significant.

Table 2

Descriptive statistics for achievement and each scale

	$\bar{X}$	SD
Achievement in computer programming language courses	67.41	22.43
Computer programming self-efficacy	3.87	.63
Perceived learning	3.61	.52
ATCP	3.73	.72

Table 3
Correlations among variables

	1	2	3	4	5	6	7
1. ATCP	-	.020	-.117	-.171	.473**	.738**	.710**
2. Grade level		-	-.062	.241**	-.176	.178	-.015
3. Gender			-	-.024	-.098	-.116	-.074
4. High school type				-	-.165	-.129	-.143
5. Achievement mean					-	.409**	.356**
6. Self-efficacy						-	.701**
7. Perceived learning							-

** Correlation is significant at the 0.05 level (1-tailed).

3.3. *Regression Analysis*

In the introduction part, depending on the findings in the literature, it was hypothesized that gender, school type and grade level have effects on the ATCP. However, they were excluded from the multiple regression analysis, because they were not significantly correlated with the dependent variable (Field, 2009). If there is a strong correlation between the predictor variables, multicollinearity, which may violate the one of the assumptions of the multiple regression, will exist. If the collinearity between the predictors is perfect, then it will be difficult to distinguish the unique estimates of regression coefficients. In this study, because the correlations between the predictors are below .80, and variance inflation factors (VIF) are below 10 (Cohen, Cohen, West, & Alken, 2003; Field, 2009), multicollinearity between predictors is not worthy of concern.

Table 4 represents the result of multiple-regression analysis conducted to calculate the explanation power of achievement in computer programming lessons, computer programming self-efficacy and perceived learning on students' ATCP. It was found that this model is significant (F = 69.108, p < .05) and accounted for 64.3% of the variance (R = .802) in pre-service computer science teachers' ATCP. According to the Table 4, perceived learning (t = 4.510), computer programming self-efficacy (t = 5.203), and achievement mean in computer programming courses (t = 2.868) were significantly positive predictors for students' ATCP. As can be seen in the Table 4, perceived-learning, computer programming self-efficacy and achievement mean have positive weights, indicating that students with higher achievement, computer programming self-efficacy and perceived learning scores were expected to have higher attitude scores. Based on this regression analysis, the equation for the regression line for predicting students' ATCP was:

$$\text{ATCP} = \text{perceived learning} \times 0.485 + \text{computer programming self-efficacy} \times 0.478 \\ + \text{achievement mean} \times 0.006 - 0.252$$

Table 4

Results of multiple linear regression analysis

Variable	B	SE	β	t	p
Constant)	-.252	.288		-.873	.385
Achievement mean	.006	.002	.176	2.868	.005
Self-efficacy	.478	.092	.418	5.203	.000
Perceived learning	.485	.108	.354	4.510	.000
R = 0.802	R^2 = 0.643		$R^2_{(adjusted)}$ = 0.634		
F = 69.108	p = 0.000				

4. Discussion and Conclusion

This study aimed to investigate the factors that affect the pre-service computer science teachers' ATCP. With this aim, students' gender, high school type, grade level, achievement in computer programming language courses, computer programming self-efficacy, and perceived learning were subjected to regression analysis in order to test whether these variables could be used to predict their ATCP. The results of the multiple-regression analysis suggested that students' academic achievement, computer programming self-efficacy and perceived learning play important role in explaining their ATCP, accounting for 64.3% of the variance explained. In contrast, high school type, grade level and gender were not significant predictors of ATCP.

There are studies investigating the relationship between affective and cognitive domains. In one of the influential papers examining the correlation between achievement and attitude, after investigating 113 primary studies, Ma and Kishor (1997) emphasized that there is a positive correlation between attitude and achievement, but this relationship is not statistically significant. On the other hand, some researchers found a significant relationship between attitude and achievement (Recber, Isiksal, & Koc, 2018). Correlation between achievement in computer programming lessons and ATCP was positive and significant (r = .473, p < .01) in the current study. In addition, academic achievement was one of the significant predictors of students' ATCP. Similar to the current study, Baser (2013b) carried out a study with 179 pre-service teachers and stated that there is a significant positive correlation between students' ATCP and their achievements in programming. Lee, Kim, and Lee (2017) conducted a study with 4221 elementary school students and also found that ATCP was highly correlated with academic achievement in programming education.

Research investigating the relationship between attitude and self-efficacy has shown that there is positive correlation between these variables (Kinzie *et al.*, 1994; Ozyurt, 2015; Ozyurt & Ozyurt, 2015). Parallel to the findings of the previous research, in this study, it was found that there was a positive and significant correlation (r = .738, p < .01) between students' computer programming self-efficacy and their ATCP. Moreover, it was found that computer programming self-efficacy was a significant variable to predict ATCP. This finding was similar to the results of previous studies (Ozyurt, 2015; Ozyurt

& Ozyurt, 2015). Contrary to current study's finding, Yagci (2016) investigated 279 pre-service teachers and couldn't find any correlation between ATCP and their computer programming self-efficacy.

Perceived learning or self-reports of learning can be used for measure of learning although there are some criticism on the validity and reliability issues (Gravestock & Gregor-Greenleaf, 2008; Rovai *et al.*, 2009; Van Sickle, 2016). While not specifically focused on computer programming, there are studies that show attitude and perceived learning of the participants are highly correlated (Lin, 2018; Wong, Kong, & Hui, 2017). Similar to the findings of these studies, current study showed that the perceived learning in computer programming was positively and highly correlated with ATCP (r = .710, p < .01). Furthermore, ATCP could be predicted using the perceived learning variable.

Students graduated from computer science related vocational high schools had several courses and experiences on computer programming. It is expected that a student who has graduated from this type of school has a different ATCP than a student who has graduated from other type of schools (Erol & Kurt, 2017). In the current study a non – significant correlation between ATCP and students' high school type was computed. There are controversial results on the effect of students' graduated high school type on ATCP in the literature. For example, in Yagci's (2016) study, there was a positive and meaningful difference in favor of graduates of vocational high school on ATCP. Karaci (2016a) found that vocational high school graduates had significantly less ATCP than other school type graduates. There are also studies that show no significant differences on ATCP according to the type of graduated high schools (Erol & Kurt, 2017; Karaci, 2016b). Erol and Kurt (2017) stated that programming education in vocational high school is inadequate and this may be the reason for the controversial results in the literature.

One of the findings of the study showed that the correlation between gender and ATCP was not statistically significant. Similar to this finding of the current study, there are various studies that show no significant difference of males' and females' ATCP (Erol and Kurt, 2017; Godbole, 2014; Korkmaz, 2016; Yagci, 2016). On the contrary, some studies show that male students have significantly higher ATCP compared to female students (Baser 2013b; Karaci 2016a; Karaci, 2016b; Ozyurt, 2015; Ozyurt & Ozyurt, 2015). Although, researchers have found gender differences in ICT areas, the gender gap has decreased over the past several years, as seen in the current study (Alsadoon, 2013; Huang, Hood, & Yoo 2013; Top, Yukselturk, & Cakir, 2011).

Another result of this study indicated that students' grade level was not significantly related to their ATCP. There are controversial results in literature considering the effect of grade level on ATCP. Godbole (2014), Yagci (2016) and Karaci (2016b) reported that higher education students' ATCP did not differ according to their grade levels. On the other hand, there are studies showing students' ATCP differ according to their grade levels; some indicating higher graders had higher ATCP (Erol & Kurt, 2017), and some showing lower graders had higher ATCP (Karaci, 2016a; Ozyurt & Ozyurt, 2015). The reason for these differences were attributed to some factors like increased number of programming courses, more sophisticated programming courses, failing in programming courses, or increased study loads of students (Karaci, 2016a).

4.1. *Conclusion*

The high school type that students graduated did not predict the attitude in the study. It might be expected that university students graduated from computer science related vocational high schools have higher ATCP scores than the university students who graduated from other high schools, since their previous experience might help them have higher grades and better learning. However, instead of starting programming education at an earlier age, the focal points should give quality education which takes into consideration the development of students' attitudes. Students who had low level quality education – that may be the case for some vocational high schools (Erol & Kurt, 2017) – in their high schools may not be able to produce expected attitudes. Another explanation may be that at the earlier stages of their university education, students' high school computer programming experience can help them to have better grades and learn better but this advantage can decrease over time and fade away. Or, it may be the case that since the instruction and expectations from students are different in universities from high schools, previous programming experience did not provide any advantage for vocational high school students.

The curriculum of the department that participants attended in the study includes programming related courses. The number, the type and the complexity of the computing courses increase gradually by grade level till fourth grade in their department. One might expect that ATCP should improve depending on the grade level due to the increasing knowledge about programming. However, grade level was not a significant predictor for ATCP in the current study. There might be two possible explanations among the others. First one is related to the participants. The participants of this study were potential computer science future teachers or they will be most likely recruited in ICT related profession. It might be the case that since their entry level ATCP was high enough, the correlation between grade level and ATCP was low. The second explanation is related with courses offered to the pre-service teachers. The increasing number and complexity of programming courses could cause cognitive difficulties that in turn saved the participants from developing more positive ATCP depending on the grade level (Karaci, 2016a).

In the literature there are many studies stating that computer science is a male dominated field, females are underrepresented in computer science related courses in universities and only small percentage of females choose computer science as a future career (Cheryan, Master, & Meltzoff, 2015; Galpin, 2002; Scragg & Smith, 1998). Nevertheless, gender was not found to be a significant factor for predicting attitude in this study. There are two possible explanations for this finding, the first of which is related to the computational thinking movement. There has been a recent interest related to computational thinking in the world (Mohaghegh & McCauley, 2016). This interest has had its place in Turkey and low level, high ceiling and wide walls programming environments (Resnick *et al.*, 2009), e.g. Scratch, are started to be used in universities (Cetin, 2016). It might be the case these programming environments helped girls to improve their computer scientist image and improved their ATCP. However, it would be early to draw out the result that gender difference is not an issue related to ATCP because of the recent instructional improvements. It will be better to test this hypothesis with different studies to get more sense. The second and more probable explanation is related to majors of the students. The students participated in the study will not be computer scientists or computer engineers; rather they will be expected to be computing teachers in middle and high schools. So, they probably considered computer programming in the context of computer science education. Unlike computer science, females do not have negative attitudes towards education as a profession (Mastekaasa & Smeby, 2008). Therefore, gender was not found to be a significant factor since participants of the study considered programming in the context of teaching programming to middle and high school students.

In the literature, there have been conflicts about the relationship between achievement and learning perception; either perceived learning or achievement scores is a good indicator of actual learning (Metcalfe, 2009; Rovai *et al.*, 2009). This study brings new insights to the literature on achievement versus perceived learning debate in the ATCP context. It was found that there was a positive and high correlation between perceived learning and achievement scores. However, multicollinearity was not existed between achievement and perceived learning. And also, they were found to be separate contributors for the prediction of ATCP. Although perceived learning and achievement scores are highly correlated, they are different constructs in the prediction of ATCP. In contrast to the literature discussing either achievement or perceived learning is more valuable to describe overall success of the individuals, results of this study suggest that the issue is not to choose between the two; rather, the issue is to use both to see a greater picture.

Positive and significant correlation between students' computer programming self-efficacy and their ATCP was found in this study and computer programming self-efficacy was a significant predictor variable for ATCP. Self-efficacy is the beliefs about a subject of the individual on his / her skill in order to be able perform a task on that subject (Bandura, 1997). It is not an extraordinary finding that individuals who feel efficient in a field have positive attitudes towards this field. However, this result should be considered cautiously. Computer programming self-efficacy and ATCP levels of pre-service teachers were at high level in this study. Since they intentionally chose their department in the university entrance exam and they will likely work in computing education related jobs after graduation, high levels of self-efficacy and ATCP were expected and intended result for the participants of the study. So, participants of this study might exclude the potential individuals who feel efficient in programming but do not want to pursue it as a profession due to some reasons, including having negative attitude towards computer programming.

The study contributes to the literature on pre-service computer science teachers' ATCP. Data presented in this study showed that the ATCP of pre-service teachers can be predicted using the achievement in computer programming courses, computer programming self-efficacy, and perceived learning. Attitude is an important variable in education. Achievement in computer programming courses, computer programming self-efficacy, and perceived learning variables can be considered to improve attitude according to the results of this study. However, the results of this study mainly depends on multiple linear regression which was conducted on the data in order to analyze the prediction effects of several variables on ATCP. One step further of this study may be to develop a model to examine the variables that have significant effect on the ATCP of students using the structural equation modelling. Moreover, 64.3% of the variance of the dependent variable, attitude toward computer programming, was explained by the four variables, leaving 35.7% unexplained. A future study may consider other variables that might have been used to predict ATCP. Additionally, further studies need to be conducted to explore the relationship among these predictor variables as a whole.

References

Aiken, L. R. (2002). *Attitudes and related psychosocial constructs: Theories, assessment, and research: theories, assessment, and research*. London: Sage Publications.

Alavi, M. (1994). Computer-mediated collaborative learning: An empirical evaluation. *MIS Quarterly*, *18*(2), 159–174.

Alsadoon, E. A. (2013). Factors influencing faculty to adopt web applications in their teaching. Unpublished PhD thesis, Ohio University, OH.

Altun, A., & Mazman, S. G. (2012). Programlamaya ilişkin öz yeterlilik algısı ölçeğinin Türkçe formumun geçerlilik ve güvenirlik çalışması [The validity and reliability study of the Turkish version of the self-efficacy perception scale for programming]. *Eğitimde ve Psikolojide Ölçme ve Değerlendirme Dergisi*, *3*(2), 297–308.

Armoni, M., Meerbaum-Salant, O., & Ben-Ari, M. (2015). From scratch to "Real" programming. *ACM Transactions on Computing Education*, *14*(4), 25.

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), Retrieved from `https://files.eric.ed.gov/fulltext/ED503900.pdf`.

Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review*, *84*(2), 191–215.

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

Baser, M. (2013a). Attitude, gender and achievement in computer programming. *Middle-East Journal of Scientific Research*, *14*(2), 248–255.

Baser, M. (2013b). Developing attitude scale toward computer programming. *International Journal of Social Science*, *6*(6), 199–215.

Bell, T., Rosamond, F., & Casey, N. (2012). Computer science unplugged and related projects in math and computer science popularization. In H. L. Bodlaender *et al.* (Eds.), *The multivariate algorithmic revolution and beyond. Lecture notes in computer science* (pp. 398–456). Berlin: Springer.

Boulay, B. D. (1986). Some difficulties of learning to program. *Journal of Educational Computing Research*, *2*(1), 57–73.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Annual American Educational Research Association meeting*, Vancouver, BC, Canada. Retrieved from `http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf`

Cetin, I. (2016). Preservice teachers' introduction to computing exploring utilization of Scratch. *Journal of Educational Computing Research*, *54*(7), 997–1021.

Cetin, I., & Andrews-Larson, C. (2016). Learning sorting algorithms through visualization construction. *Computer Science Education*, *26*(1), 27–43.

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

Cetin, I., Sendurur, E., & Sendurur, P. (2014). Assessing the impact of meta-cognitive training on students' understanding of introductory programming concepts. *Journal of Educational Computing Research, 50*(4), 507–524.

Cheryan, S., Master, A., & Meltzoff, A. N. (2015). Cultural stereotypes as gatekeepers: Increasing girls' interest in computer science and engineering by diversifying stereotypes. *Frontiers in psychology*, *6*, 49.

Retrieved from https://www.frontiersin.org/articles/10.3389/fpsyg.2015.00049/full

Cohen, J., Cohen, P., West, S. G., & Alken, L.S. (2003). *Applied multiple regression/correlation analysis for the behavioral sciences. 3rd Edition*. New York: Lawrence Erlbaum Associates.

Cukier, W., Shortt, D., & Devine, I. (2002). Gender and information technology: Implications of definitions. *ACM SIGCSE Bulletin*, *34*(4), 142–148.

Department for Education (2014). National curriculum in England: Framework for key stages 1 to 4. Retrieved from https://www.gov.uk/government/publications/national-curriculum-in-england-framework-for-key-stages-1-to-4/the-national-curriculum-in-england-framework-for-key-stages-1-to-4

Doube, W., & Lang, C. (2012). Gender and stereotypes in motivation to study computer programming for careers in multimedia. *Computer Science Education*, *22*(1), 63–78.

Erol, O., & Kurt, A. A. (2017). Investigation of CEIT Students' Attitudes towards Programming. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, *1*(41), 314–325.

Field, A. (2009). *Discovering statistics with SPSS, 3rd Edition.* California: Sage Publications.

Galpin, V. (2002). Women in computing around the world. *ACM SIGCSE Bulletin*, *34*(2), 94–100.

Godbole, S. (2014). Impact of a visual programming experience on the attitude toward programming of introductory undergraduate students. Unpublished Master's thesis, Purdue University, IN.

Götschi, T., Sanders, I., & Galpin, V. (2003). Mental models of recursion. *ACM SIGCSE Bulletin*, *35*(1), 346–350.

Gravestock, P., & Gregor-Greenleaf, E. (2008). *Student course evaluations: Research, models and trends*. Toronto: Higher Education Quality Council of Ontario.

Gurer, D., & Camp, T. (2002). An ACM-W literature review on women in computing. *ACM SIGCSE Bulletin*, *34*(2), 121–127.

Huang, W.-H. D., Hood, D. W., & Yoo, S. J. (2013). Gender divide and acceptance of collaborative Web 2.0 applications for learning in higher education. *The Internet and Higher Education*, *16*, 57–65.

Hubwieser, P., Armoni, M., Giannakos, M. N., & Mittermeir, R. T. (2014). Perspectives and visions of computer science education in primary and secondary (K-12) schools. *ACM Transactions on Computing Education*, *14*(2), 7, doi:10.1145/2602482

Karaci, A. (2016a). Investigation of attitudes towards computer programming in terms of various variables. *International Journal of Programming Languages and Applications*, *6*(1/2), doi:10.5121/ijpla.2016.6201

Karaci, A. (2016b). Uzaktan eğitim öğrencilerinin bilgisayar programlamaya yönelik tutumlarının incelenmesi [Examining the attitudes of distance education students towards computer programming]. *Proceedings of the ERPA International Congresses on Education Conference* (pp. 738–744). Sarajevo/Bosnia and Herzegovina.

Kinzie, M. B., Delcourt, M. A., & Powers, S. M. (1994). Computer technologies: Attitudes and self-efficacy across undergraduate disciplines. *Research in higher education*, *35*(6), 745–768.

Koriat, A., & Bjork, R. A. (2005). Illusions of competence in monitoring one's knowledge during study. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *31*(2), 187–194.

Korkmaz, O. (2016). The effects of Scratch-based game activities on students' attitudes, self-efficacy and academic achievement. *International Journal of Modern Education and Computer Science*, *8*(1), 16–23.

Korkmaz, O., & Altun, H. (2013). Engineering and CEIT student's attitude towards learning computer programming. *International Journal of Social Science*, *6*(2), 1169–1185.

Lee, S., Kim, J., & Lee, W. (2017). Analysis of factors affecting achievement in maker programming education in the age of wireless communication. *Wireless Personal Communications*, *93*(1), 187–209.

Lin, G.-Y. (2018). Anonymous versus identified peer assessment via a Facebook-based learning application: Effects on quality of peer feedback, perceived learning, perceived fairness, and attitude toward the system. *Computers & Education*, *116*, 81–92.

Ma, X., & Kishor, N. (1997). Assessing the relationship between attitude toward mathematics and achievement in mathematics: A meta-analysis. *Journal for Research in Mathematics Education*, *28*(1), 26–47.

Mastekaasa, A., & Smeby, J. C. (2008). Educational choice and persistence in male-and female-dominated fields. *Higher Education*, *55*(2), 189–202.

Metcalfe, J. (2009). Metacognitive judgments and control of study. *Current Directions in Psychological Science*, *18*(3), 159–163.

Mohaghegh, M., & McCauley, M. (2016). Computational thinking: The skill set of the 21st century. *International Journal of Computer Science and Information Technologies*, *7*(3), 1524–1530.

Norris, P. (2001). *Digital Divide: Civic engagement, information poverty, and the Internet worldwide*. NY: Cambridge University Press.

Ozyurt, O. (2015). An analysis on distance education computer programming students' attitudes regarding programming and their self-efficacy for programming. *Turkish Online Journal of Distance Education, 16*(2), 111–121.

Ozyurt, O., & Ozyurt, H. (2015). A study for determining computer programming students' attitudes towards programming and their programming self-efficacy. *Journal of Theory & Practice in Education*, *11*(1), 51–67.

Ramalingam, V., & Wiedenbeck, S. (1998). Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self-efficacy. *Journal of Educational Computing Research*, *19*(4), 367–381.

Ragonis, N., & Ben-Ari, M. (2005). A long-term investigation of the comprehension of OOP concepts by novices. *Computer Science Education*, *15*(3), 203–221.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., & Kafai, Y. (2009). Scratch: Programming for all. *Communications of the ACM*, *52*(11), 60–67.

Recber, S., Isiksal, M., & Koc, Y. (2018). Investigating self-efficacy, anxiety, attitudes and mathematics achievement regarding gender and school type. *Anales de Psicología*, *34* (1), 41–51.

Rovai, A. P., Wighting, M. J., Baker, J. D., & Grooms, L. D. (2009). Development of an instrument to measure perceived cognitive, affective, and psychomotor learning in traditional and virtual higher education classroom settings. *Internet and Higher Education*, *12*(1), 7–13.

Scragg, G., & Smith, J. (1998). A study of barriers to women in undergraduate computer science. *ACM SIGCSE Bulletin*, *30*(1), 82–86.

Seehorn, D., Carey, S., Fuschetto, B., Lee, I., Moix, D., O'Grady-Cunniff, D., *et al.* (2011). *CSTA K–12 computer science standards*. New York: ACM/CSTA.

Selby, C., & Woollard, J. (2013). Computational thinking: The developing definition. In J. Carter, I. Utting, & A. Clear (Eds.), *Proceedings of 18th annual conference on innovation and technology in computer science education* (p. 6). Canterbury: University of Southampton. Retrieved from `https://eprints.soton.ac.uk/356481/1/Selby_Woollard_bg_soton_eprints.pdf`

Singh, K., Allen, K. R., Scheckler, R., & Darlington, L. (2007). Women in computer-related majors: A critical synthesis of research and theory from 1994 to 2005. *Review of Educational Research*, *77*(4), 500–533.

Stoilescu, D., & Egodawatte, G. (2010). Gender differences in the use of computers, programming, and peer interactions in computer science classrooms. *Computer Science Education, 20*(4), 283–300.

Top, E., Yukselturk, E., & Cakir. R. (2011). Gender and Web 2.0 technology awareness among ICT teachers. *British Journal of Educational Technology*, *42*(5), E106-E109.

Top, E., Yukselturk, E., & Inan, F. A. (2010). Reconsidering usage of blogging in preservice teacher education courses. *The Internet and Higher Education*, *13*(4), 214–217.

Trilling, B., & Fadel, C. (2009). *21st century skills: Learning for life in our times*. San Francisco, CA: Jossey-Bass.

Van Sickle, J. R. (2016). Discrepancies between student perception and achievement of learning outcomes in a flipped classroom. *Journal of the Scholarship of Teaching and Learning*, *16*(2), 29–38.

Wiedenbeck, S., Fix, V., & Scholtz, J. (1993). Characteristics of the mental representations of novice and expert programmers: An empirical study. *International Journal of Man-Machine Studies*, *39*(5), 793–812.

Winberg, T. M., & Hedman, L. (2008). Student attitudes toward learning, level of pre-knowledge and instruction type in a computer-simulation: Effects on flow experiences and perceived learning outcomes. *Instructional Science*, *36*(4), 269–287.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33–35.

Wing, J. M. (2011, February). Research notebook: Computational thinking- what and why? *The Link Magazine*, 20–23. Retrieved from `https://www.scs.cmu.edu/link.`

Wong, E. Y.-C., Kong, K. H., & Hui, R. T.-Y. (2017). The influence of learners' openness to IT experience on the attitude and perceived learning effectiveness with virtual reality technologies. In *Proceedings of the 6th IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)*, (pp. 118–123). doi:10.1109/TALE.2017.8252315

Wu, Y. T., & Tsai, C. C. (2006). University students' internet attitudes and internet self-efficacy: A study at three universities in Taiwan. *CyberPsychology & Behavior*, *9*(4), 441–450.

Yagci, 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. *Journal of Human Sciences*, *13*(1), 1418–1432.

M.D. Gurer works as an Assistant Professor in the Computer Education and Instructional Technology Department in Bolu Abant Izzet Baysal University in Turkey. He offers courses such as Computer Literacy, Using Technology in Education, Distance Education, Research Methods, and Introduction to Statistics both at BA and MA levels. His research interests include Web 2.0 tools in education, distance education, and technology integration in learning environments.

I. Cetin received his PhD degree in Computer Education and Instructional Technologies Department in Middle East Technical University, Ankara, Turkey in 2009. Currently he is an associated professor in Computer Education and Instructional Technology Department in Bolu Abant Izzet Baysal University. He is interested in mathematical thinking, computational thinking, and the use of technology (e.g. visualization, games and computer algebra systems) in computer science and mathematics education

E. Top completed his PhD in the Department of Computer Education and Instructional Technology in the Middle East Technical University (METU), Turkey. He currently works at the Department of Computer Education and Instructional Technology in Bolu Abant Izzet Baysal University, Bolu, Turkey. His research interests include computational thinking, online course design, team based learning, educational technology integration processes, Web 2.0 technologies, and assessment and evaluation processes.