Journal of Theoretical Educational Sciences / Kuramsal Eğitimbilim Dergisi
Volume 19 Issue 1, January 2026, pp. 123-145
e-ISSN: 1308-1659 | DOI: http://doi.org/10.30831/akukeg.1757456

Examining the Factors Affecting University Students' Perceived Learning Levels in Programming*

Üniversite Öğrencilerinin Programlamaya İlişkin Algılanan Öğrenme Düzeylerini Yordayan Faktörlerin İncelenmesi

Gamze USLU** **Erhan ÜNAL*****

Received: 4 August 2025 **Research Article** **Accepted:** 19 November 2025

ABSTRACT: This study explores the relationship between university students' attitudes toward programming, their motivation, anxiety levels, self-efficacy perceptions, and how these factors influence their perceived learning in programming. A correlational research method was employed, and data were collected from 207 university students. After analysing the data for normality and checking the assumptions for multiple regression analysis, the responses of 10 participants were excluded from the dataset. As a result, the analysis was conducted using data from 197 students. Data were collected through online forms, using several instruments: a demographic information form, the Computer Programming Attitude Scale, the Learning Motivation in Computer Programming Courses Scale, the Computer Programming Self-Efficacy Scale for Computer Literacy Education, the Programming Anxiety Scale, and the Perceived Learning Scale. The study reveals that university students possess a strong attitude, motivation, self-efficacy, and perceived learning in programming, while their anxiety levels are moderate. A strong positive relationship was found between students' perceived learning levels in programming and their self-efficacy perceptions. Additionally, perceived learning moderately and positively correlated with students' motivation and attitudes toward programming. Conversely, a moderate negative correlation was found between perceived learning and programming-related anxiety. Ultimately, students' perceived learning levels in programming were predicted by their self-efficacy and motivation.

Keywords: Programming, motivation, attitude, self-efficacy, anxiety, perceived learning.

ÖZ: Bu araştırmanın amacı, üniversite öğrencilerinin programlamaya yönelik tutum, motivasyon, kaygı, öz yeterlik algıları ile programlamaya ilişkin algılanan öğrenme düzeyleri arasındaki ilişkiyi belirlemek ve bu değişkenlerin programlamaya ilişkin algılanan öğrenme düzeylerini yordama düzeyini saptayabilmektir. Araştırma korelasyonel araştırma modeline göre tasarlanmış olup 207 üniversite öğrencisinden veri toplanmıştır. Normallik ve çoklu regresyon analizi varsayımları kontrolü sonucunda 10 katılımcı verisi veri setinden çıkartılmıştır. Sonuç olarak 197 öğrencinin verisi ile analizler gerçekleştirilmiştir. Araştırmada veriler demografik bilgi formu, "Bilgisayar Programlama Tutum Ölçeği", "Bilgisayar Programlama Derslerinde Öğrenme Motivasyonu Ölçeği", "Bilgisayar Okuryazarlığı Eğitimi için Bilgisayar Programlama Öz Yeterlik Ölçeği", "Bilgisayar Programlama Kaygı Ölçeği", "Algılanan Öğrenme Ölçeği" kullanılarak çevrim içi form aracılığıyla toplanmıştır. Yapılan analizler sonucunda üniversite öğrencilerinin programlamaya karşı tutum, motivasyon, öz yeterlik algıları ve algılanan öğrenmelerinin yüksek düzeyde, programlamaya karşı kaygılarının ise orta düzeyde olduğu ortaya çıkmıştır. Ayrıca programlamaya karşı algılanan öğrenmenin öz yeterlik algısı ile yüksek düzeyde; programlamaya karşı tutum ve motivasyon değişkenleri ile orta düzeyde pozitif ve anlamlı bir ilişkiye sahip olduğu görülürken, kaygı değişkeni ile negatif ve anlamlı bir ilişkiye sahip olduğu tespit edilmiştir. Son olarak programlamaya karşı motivasyon ve öz yeterlik algısı değişkenlerinin programlamaya karşı algılanan öğrenmeyi anlamlı bir şekilde yordadığı bulunmuştur.

Anahtar kelimeler: Programlama, motivasyon, tutum, öz-yeterlik, kaygı, algılanan öğrenme.

* This article was produced from the master thesis prepared by the first author under the supervision of the second author.

** Instructor, Selçuk University, Konya, Türkiye, gamze.uslu@seulcuk.edu.tr, https://orcid.org/0009-0005-2170-7165
***_Corresponding Author_: Assoc. Prof. Dr., Afyon Kocatepe University, Afyonkarahisar, Türkiye, eunal@aku.edu.tr, https://orcid.org/0000-0002-5349-4193

Citation Information
Uslu, G., & Ünal, E. (2026). Examining the factors affecting university students' perceived learning levels in programming. *Journal of Theoretical Educational Sciences [Kuramsal Eğitimbilim Dergisi], 19*(1), 123-145. http://doi.org/10.30831/akukeg.1757456

Copyright © 2026 Author(s).
This is an open access article distributed under the terms of the CC BY-NC-SA 4.0 license.

Software and other information systems are key to our lives as they can automate processes, enhance communication, improve products and services, and boost productivity (Santos et al., 2020). As a result, people are more dependent on software. According to the Organisation for Economic Co-operation and Development (OECD) (2024), the information and communication technologies (ICT) sector has grown approximately three times faster than the overall economy of OECD countries over the past decade. In 2023, this sector achieved a remarkable growth rate of 7.6%. Thus, younger generations must learn to develop new software instead of only consuming those developed by others in this era of rapid advancement (Demirer & Sak, 2016). Programming skills may contribute to developing skills and abilities related to 21st-century skills and digital proficiency (Mahmud & Wong, 2022). According to the Computer Science Teachers Association (CSTA) and the International Society for Technology in Education (ISTE), individuals in the information age need to understand how, where, and when to use computers and digital tools for effective problem-solving (CSTA & ISTE, 2011). Providing programming education to students and nurturing their programming skills is key to achieving this goal.

Many countries have started incorporating programming or coding courses into their curricula to help students gain programming skills (Bocconi et al., 2016). According to a European Schoolnet report (2015), 16 European countries have integrated programming into their educational frameworks, and two countries plan to integrate it. This inclusion is motivated by factors such as improving 21st-century skills, enhancing problem-solving abilities, fostering coding skills, attracting more students to pursue computer science, and supporting other essential competencies. The Eurydice Report (2022) also highlights that several countries, including Estonia, Greece, France, Croatia, Cyprus, Latvia, Hungary, Poland, Slovakia, Finland, Sweden, Switzerland, Liechtenstein, North Macedonia, Serbia, and Turkey, have established learning outcomes related to programming in their curricula across all education levels, from primary to secondary education. At the higher education level, students enroll in programming courses in informatics. Students in non-computing disciplines also enroll in programming-related courses (Ayalew et al., 2018). These courses allow students to cultivate coding, problem-solving, and computational thinking competencies before their careers. By engaging with programming, students learn computational thinking, which encompasses more than just writing code—it fosters important problem-solving abilities (Lye & Koh, 2014).

Programming includes writing computer programs through computational thinking effectively (Lye & Koh, 2014; Mills et al., 2024) and is essential to developing 21st-century abilities for both individuals. A variety of studies demonstrate that programming enhances key cognitive skills such as problem-solving and computational thinking (Fessakis et al., 2013; Gürbüztürk & Yılmaz-Tanataş, 2024; Yıldız & Seferoğlu, 2021). In learning programming, individuals need to solve problems and learn the programming language's syntax and key programming concepts (Mannila et al., 2006). Rogerson and Scott (2010) highlight several challenges in teaching programming, mainly due to its inherent nature and complexity. Programming is a complex and challenging subject, often leading to failures and frustration for many students during their learning process (Erol & Kurt, 2017). A study by Krpan et al. (2015) revealed that around 46% of 511 students who enrolled in an introductory

programming course from 2010 to 2013 failed to pass. These negative outcomes have prompted researchers to explore new methodologies and tools to support students (Gomes & Mendes, 2007).

Programming courses are generally perceived as difficult (Aşkar & Davenport, 2009; Bergin & Reilly, 2005; Gomes & Mendes, 2007), and students' success rates tend to be low (Robins et al., 2003). Introductory programming courses have high rates of dropouts and failures at higher education institutions, indicating that learning programming can be quite challenging (Olelewe & Agomuo, 2016). Various factors influencing programming skills have been examined, including attitudes toward programming, motivation, self-efficacy, and anxiety. The difficulties students encounter while learning programming can lead to the development of negative attitudes (Başer, 2013b), a decrease in motivation (Baştemur-Kaya, 2018), and fear of programming (Rogerson & Scott, 2010). These challenges can also diminish their belief in their ability to succeed, thereby reducing their self-efficacy (Altun & Mazman, 2012) and ultimately resulting in failure. Considering the significance of programming in providing people with critical 21st-century skills, scholars are beginning to investigate methods of teaching programming and overcome challenges. Thus, one research topic is examining factors, including motivation, satisfaction, and attitudes toward programming courses, to predict programming success (Tokumacı, 2019). Programming skill is associated with several key factors such as attitude (Başer, 2013a; Hongwarittorrn & Krairit, 2010; Tokumacı, 2019), self-efficacy level (Mazman & Altun, 2013; Özmen & Altun, 2014; Ramalingam et al., 2004), motivation (Erol, 2015; Gomes & Mendes, 2007), anxiety (Rogerson & Scott, 2010; Yıldırım, 2022), and programming experience (Ramalingam et al., 2004; Rountree et al., 2004; Yılmaz & Çakır, 2019).

Problem Statement

Computer programming is essential skill across industries (Mutanga, 2020; Yılmaz & Yılmaz, 2023) Those who know programming can develop digital tools and technologies that drive economic growth (Yılmaz & Yılmaz, 2023). According to ISTE Standards for Students (2016), students, as computational thinkers, effectively use technology to analyze and solve problems through strategic approaches. With the increasing global focus on programming education and its integration into curricula at all levels, understanding the factors that shape students' learning in programming has become essential. As stated earlier, there are several factors influencing students' programming performance. While numerous studies have investigated the determinants of students' achievement in programming, their findings are often inconsistent due to differences in context, culture, learner and instructor characteristics, instructional environments and other factors (Ayalew et al., 2018). These inconsistencies indicate a need for further empirical evidence to clarify which factors most significantly influence students' learning outcomes in programming. This study aims to identify the key factors affecting students' perceived learning levels in programming at the higher education level. Thus potential factors related to programming identified in the literature include attitude, motivation, self-efficacy, and anxiety. In doing so, it seeks to contribute to the literature in both computer science and educational sciences, providing insights into effective programming education. The findings are expected to offer valuable guidance for researchers and educators in designing learning environments and instructional

strategies that enhance students' engagement, confidence, and overall success in programming.

Literature Review

Attitudes toward Programming

Attitude is one of the factors affecting programming skills and achievement. According to Breckler (1984, p. 1191), "attitude is defined as a response to an antecedent stimulus or attitude object. The stimulus may or may not be observable and can best be thought of as an independent or exogenous variable". Three main types of responses to this stimulus are affect, behaviour, and cognition (Breckler, 1984). The cognitive component reflects individuals' beliefs, knowledge, or thoughts about a particular object or situation. The affective component represents their feelings toward that object or situation, while the behavioural component indicates their tendency to act in relation to it (Breckler, 1984; Reich-Stiebert et al., 2019). These three interconnected components influence how a person will respond to the object or situation (Reich-Stiebert et al., 2019). Similarly, Çetin and Özden (2015) analyzed programming attitudes across affection, cognition, and behaviour. Thus, students' academic success in any course relates to positive or negative attitudes (Amnouychokanant et al., 2021). Başer (2013a) concluded that there was a significant positive relationship between students' programming achievement and their attitudes towards programming. Moreover, Amnouychokanant et al. (2021) found that students with higher academic achievement held significantly more positive attitudes toward programming, whereas those with lower achievement exhibited less favorable perceptions. A literature review reveals several studies examining the effects of various factors on students' attitudes towards programming (Erol & Kurt, 2017; Günbatar, 2018; Gürer et al., 2019; Karaci, 2016). Therefore, university students' attitudes towards programming can influence their perceived learning level in programming.

Motivation

Motivation is another psychological factor influencing programming skills, which refers to initiating and maintaining goal-directed action (Pintrich & Schunk, 1996). Schunk (2012) asserts that motivation increases when a student perceives the possibility of success in a subject, while it decreases in the face of potential failure. Yağcı (2016b) notes that a student's attitude toward programming, the complexity of the programming language, and motivation all play critical roles in programming success. Furthermore, Çilengir and İzmirli (2023) argue that when students perceive programming as abstract or when suitable teaching methods are not employed, students' motivation to learn programming may diminish. Özonur (2022) found that computer programming students exhibit high learning motivation in their courses. Additionally, Yılmaz and Çakır (2019) concluded that factors such as gender, type of high school attended, overall academic grade point average, and motivation significantly predict programming success. As a result of the literature, motivation is considered a crucial factor related to programming skills.

Self-efficacy

Self-efficacy refers to "people's beliefs about their capabilities to produce designated levels of performance that exercise influence over events that affect their lives." (Bandura, 1994, p. 71). Research shows that students with low self-efficacy are more likely to discontinue their academic work than those with strong self-efficacy. (Horzum & Çakır 2009). Moreover, individuals with strong self-efficacy beliefs are more resilient when facing challenges (Akkoyunlu & Kurbanoğlu 2004; Compeau & Higgins 1995; Özyurt & Özyurt 2015). Therefore, students must have a strong belief in self-efficacy regarding programming to overcome challenges. Several studies concerning self-efficacy in programming were conducted (Altun & Mazman; 2012; Kanaparan et al., 2017; Mazman & Altun, 2013). Ramalingam et al. (2004) also emphasized the significance of self-efficacy as a key factor influencing programming outcomes. Thus, the current study selects the self-efficacy variable as a predictor variable.

Anxiety

According to Mayer (2008), anxiety is an immediate mind-body response triggered by a stimulus that requires a reaction. This response has simultaneous physiological, psychological, and behavioural effects. Students who experience anxiety disorders often adopt a passive approach to learning, demonstrating a lack of interest, performing poorly on exams, and failing to complete homework (Vitasari et al., 2010). Programming anxiety can also stem from inaccurate self-assessments of one's capacity in acquiring computer programming (Connolly et al., 2007) which can hinder students from actively engaging in programming, developing solutions to problems, and forming mental schemas (Demir, 2015). Vitasari et al. (2010) found that students with high levels of anxiety tend to have low academic performance. Additionally, Keskinkılıç and Kalelioğlu (2024), Olipas et al. (2021), Şişman and Küçük (2018) investigated the programming anxiety. Thus, programming anxiety may influence university students' perceived learning level in programming.

Perceived Learning

An individual's feelings and beliefs about their learning experiences can be defined as perceived learning (Caspi & Blau, 2008). According to Johnson et al. (1996), perceived learning involves the learner's self-evaluation of the knowledge and skills acquired during the learning process (as cited in Batista & Cornachione, 2005). Caspi and Blau (2008) further explain that perceived learning is a retrospective evaluation of the learning experience. Perceived learning refers to the learner's self-report on knowledge acquisition, typically based on reflection (Bacon, 2016). Research shows that the variable of perceived learning is discussed in studies related to programming (Gürer et al., 2019; Gürer & Tokumacı, 2020b; Rebuta et al., 2022). According to Batista and Cornachione (2005), perceived learning can be a valid criterion for measuring learning success. Therefore, perceived learning of programming is selected to address students' programming levels in the current study. The perceived learning levels of university students regarding programming have been examined more comprehensively, considering variables such as attitude towards programming, motivation, self-efficacy, and anxiety.

Purpose of the Study

This study explores the relationship between university students' attitudes towards programming, motivation, anxiety levels, self-efficacy perceptions, and how these factors influence their perceived learning levels in programming. Specifically, the study seeks to address the following questions:

1. What are university students' attitudes, motivations, anxieties, self-efficacy perceptions, and perceived learning levels regarding programming?

2. Is there a significant relationship between university students' attitudes towards programming, their motivation, anxiety levels, self-efficacy perceptions, and their perceived learning regarding programming?

3. Do university students' attitudes towards programming, motivation, anxiety levels, and self-efficacy perceptions significantly predict their perceived learning levels regarding programming?

Method

Research Model

The correlational research method was employed to examine predictors of the dependent variable (Fraenkel et al., 2012). One of the purposes of correlational research is to make predictions. When a significant relationship is observed between two variables, one can estimate or predict the score of one variable based on the score of the other (Creswell, 2012; Fraenkel et al., 2012). Thus, the independent variables included attitude towards computer programming, motivation, anxiety, and self-efficacy perception, while the dependent variable was the perceived learning level in programming.

Participants

The participants are students enrolled at a state university in the Central Anatolia Region of Türkiye during the spring semester of the 2023-2024 academic year who have completed at least one programming course. Data were collected from 207 university students voluntarily. The sample selection was made using a convenience sampling method. After analysing the data for normality and checking the assumptions of multiple regression analysis, 10 participants' responses were excluded from the dataset. Consequently, data from 197 students were used for analysis. The demographic information of the university students who participated in the study is presented in Table 1.

Table 1

Demographic Distribution of University Students

Demographic	Category	*N*	Percentage
Gender	Female	92	46.7
	Male	105	53.3
Department	Information Security Technology	41	20.8

Table 1

Demographic Distribution of University Students

Demographic	Category	*N*	Percentage
	Health Information Systems Techniques	31	15.7
	Management Information Systems	125	63.5
Number of Programming Courses Taken	1	31	15.7
	2	28	14.2
	3 or more	138	70.1

According to Table 1, 53.3% of the 197 university students who participated in the study were male (n=105) and 46.7% (n=92) were female. Among these students, 63.5% (n=125) were majoring in Management Information Systems, 20.8% (n=41) in Information Security Technology, and 15.7% (n=31) in Health Information Systems Techniques. Additionally, 15.7% (n=31) of the students had taken one computer programming course, 14.2% (n=28) had taken two courses, and 70.1% (n=138) had taken three or more courses.

Data Collection Tools

The study utilized a data collection tool consisting of six sections, explained as follows. A demographic information form was employed to gather data on participants' gender, department, and the number of programming courses completed.

The Computer Programming Attitude Scale, developed by Çetin and Özden (2015), comprises three sub-dimensions and 18 items rated on a 5-point Likert scale (1 represents “strongly disagree” to 5 signifies “strongly agree”). In the current study, the reliability values were found to be 0.86 for the overall scale, 0.83 for the "affection" sub-factor, 0.69 for the "cognition" sub-factor, and 0.76 for the "behaviour" sub-factor.

The Learning Motivation in Computer Programming Courses Scale, adapted into Turkish by Avcı and Ersoy (2018), consists of six sub-dimensions and 19 items rated on a 6-point Likert scale (1 represents "totally disagree" and 6 indicates "totally agree"). The reliability and validity study revealed a Cronbach's alpha value of 0.87 for the overall scale. The alpha values for each sub-dimension are as follows: 0.73 for the "individual attitude and expectation" factor, 0.77 for the "challenging goals" factor, 0.69 for the "clear direction" factor, 0.84 for the "reward and recognition" factor, 0.72 for the "punishment" factor, and 0.79 for the "social pressure and competition" factor.

The Computer Programming Self-Efficacy Scale for Computer Literacy Education was adapted into Turkish by Gökoğlu (2022). This scale consists of three sub-dimensions and includes 16 items measured using a 5-point Likert scale, (1 represents "not like me at all" and 5 signifies "very much like me"). Regarding reliability, the calculated Cronbach's alpha for the overall scale was 0.91. Additionally, the alpha values for the individual factors are as follows: 0.89 for the "programming competence" factor, 0.81 for the "programming skills" factor, and 0.77 for the "self-regulation" factor.

The Programming Anxiety Scale, created by Yıldırım and Özdener in 2022, comprises two sub-dimensions and includes 11 items rated on a 5-point Likert scale (1 represents "never true" and 5 denotes "always true"). The overall Cronbach's alpha value for the scale is 0.91, with a value of 0.87 for the "classmates" factor and 0.87 for the "self-confidence" factor.

The Perceived Learning Scale, developed by Horzum et al. (2015), consists of five items and is structured as a single sub-dimension using a 5-point Likert scale (1 indicates strongly disagree, 5 represents strongly agree). The Cronbach's alpha value was 0.91.

The evaluation of university students' attitudes, self-efficacy, anxiety, and perceived learning levels in programming utilized five rating intervals. The scores were categorized as follows: very low (1.00–1.80), low (1.81–2.60), moderate (2.61–3.40), high (3.41–4.20), and very high (4.21–5.00). Additionally, students' motivation toward programming was assessed using a six-point scale. Scores ranging from 1.00 to 3.50 indicated negative motivation, while scores between 3.51 and 6.00 indicated positive motivation.

Data Analysis

Assessing the normal distribution assumption was the first stage in the data analysis. Thus, the normal distribution was evaluated using Skewness and Kurtosis values, which showed that they were within the range of ±2, suggesting a normal distribution (George & Mallery, 2010). Additionally, Q-Q plots and boxplots were analyzed, highlighting seven outliers in the dataset. Consequently, these outliers were removed from the dataset.

In the context of multiple regression analysis, several assumptions were checked, including the use of at least an equal interval scale for the variables, the linearity of relationships, the outlier analysis, multicollinearity, and multivariate normal distribution (Büyüköztürk, 2024). The outlier analysis assessed Mahalanobis distance values, excluding one case from the dataset. Additionally, two cases were identified as outliers through case wise diagnostics and were removed. Consequently, all remaining assumptions were satisfied, and the analysis continued with data from 197 participants.

Descriptive analyses assessed university students' attitudes, motivation, self-efficacy perception, anxiety, and perceived learning levels in programming. Pearson Correlation analysis was used to explore the relationships between students' attitudes toward programming, motivation, self-efficacy perception, anxiety levels, and their perceived learning levels in programming. Additionally, multiple regression analysis was performed to identify which variables predicted university students' perceived learning levels in programming.

Ethical Procedures

We first applied for permission from the ethics committee to conduct the study. Afyon Kocatepe University granted the required approval under ethics committee approval number 258740 on 18.03.2024. Moreover, students were informed about the purpose and scope of the study before participating. They were assured that their involvement was voluntary and that their data would be used solely for research

purposes. After providing their consent, participants completed the data collection instrument through online forms.

Results

University Students' Attitudes towards Programming, Motivation, Anxiety, Self-Efficacy, and Perceived Learning Levels in Programming

Table 2 presents descriptive statistics regarding attitudes towards programming, motivation, self-efficacy, anxiety, and perceived learning levels, including their sub-dimensions.

The descriptive statistics for the variables in this study are presented in Table 2.

Table 2

Descriptive Statistics of the Variables

Factors	*N*	$\bar{X}$	*SS*	*Skewness*	*Kurtosis*
Computer Programming Attitude	197	4.03	0.55	-0.340	-0.674
Affection	197	3.88	0.79	-0.685	0.081
Cognition	197	4.40	0.54	-0.979	0.573
Behavior	197	3.80	0.74	-0.316	-0.628
Learning Motivation in Computer Programming Courses	197	4.35	0.77	-0.042	-0.467
Individual attitude and expectation	197	4.85	0.86	-0.317	-0.852
Challenging goals	197	4.21	1.12	-0.338	-0.204
Clear direction	197	4.91	0.91	-0.542	-0.544
Reward and recognition	197	4.96	1.11	-1.143	0.791
Punishment	197	2.93	1.62	0.383	-1.019
Social pressure and competition	197	3.78	1.28	-0.158	-0.508
Computer Programming Self-Efficacy	197	3.81	0.66	-0.161	-0.526
Programming competence	197	4.03	0.78	-0.544	-0.451
Programming skills	197	3.34	0.79	0.203	-0.285
Self-regulation	197	3.99	0.75	-0.463	-0.404
Programming Anxiety	197	2.84	0.92	0.010	-0.335
Classmates	197	2.85	1.12	0.125	-0.763
Self-confidence	197	2.84	0.90	-0.077	-0.207
Perceived Learning	197	3.88	0.72	-0.284	-0.371

Table 2 shows that the affective factor, one of the sub-dimensions of the attitude scale, has a high average score of $\bar{X}$=3.88. The cognitive factor is rated even higher, with an average score of $\bar{X}$=4.40, indicating a very high level of attitude. The behavioural factor also shows a high level of attitude, with an average score of $\bar{X}$=3.80. According to the whole scale, the overall average attitude of university students towards programming is $\bar{X}$=4.03, which is classified as high.

The following results were observed among the sub-dimensions of the motivation scale: the motivation associated with the attitude and expectation factor had a mean score of $\bar{X}$=4.85, indicating a high and positive motivation level. The motivation linked to challenging goals had a mean score of $\bar{X}$=4.21, which is also high and positive. For the clear direction factor, the mean score was $\bar{X}$=4.91, reflecting a high and positive motivation. The reward and recognition factor scored the highest, with a mean of $\bar{X}$=4.96, indicating strong positive motivation. In contrast, the motivation related to the punishment factor was lower, with a mean score of $\bar{X}$=2.93, indicating low and negative motivation. The motivation associated with social pressure and competition had a moderate and positive score of $\bar{X}$=3.78. Considering the entire scale, the average motivation level of university students toward programming is $\bar{X}$=4.35, indicating a high and positive level.

Regarding the self-efficacy perception levels of programming, the findings are as follows: The programming competence factor has a mean score of $\bar{X}$=4.03, indicating a high level of self-efficacy. The programming skill factor has a mean score of $\bar{X}$=3.34, reflecting a medium level of self-efficacy. The self-regulation factor has a mean score of $\bar{X}$=3.99, indicating a high level of self-efficacy. Considering the entire scale, the average self-efficacy level of university students regarding programming is $\bar{X}$=3.81, categorizing it as high.

The overall programming anxiety levels of university students ($\bar{X}$=2.84), including classmates anxiety ($\bar{X}$=2.85) and self-confidence anxiety ($\bar{X}$=2.84), were found to be moderate. Lastly, university students' perceived programming learning levels are high, with a mean of $\bar{X}$=3.88.

Relationship between University Students' Attitudes Toward Programming, Motivation, Self-Efficacy, Anxiety Levels, and Perceived Learning Levels in Programming

Pearson correlation analysis was conducted to examine the relationship between university students' attitudes towards programming, motivation, anxiety, self-efficacy perceptions, and their perceived learning levels in programming. The findings are presented in Table 3.

Table 3

Pearson Correlation Analysis Results

	Variable	1	2	3	4	5
1	Perceived Learning Levels in Programming	1				
2	Computer Programming Attitude	0.417**	1			
3	Learning Motivation in Computer Programming Courses	0.474**	0.626**	1		
4	Computer Programming Self-Efficacy	0.705**	0.530**	0.492**	1	
5	Programming Anxiety	-0.303**	-0.418**	-0.141*	-0.443**	1

Note. **p<0.01; *p<0.05

The correlation coefficient (r) between the variables indicates a low-level correlation from 0.00 to 0.30, a medium-level correlation from 0.30 to 0.70, and a high-level correlation from 0.70 to 1.00 (Büyüköztürk, 2023). Based on this information, an analysis of Table 3 revealed a strong positive correlation (r=0.705, p<0.01) between perceived learning levels in programming and self-efficacy. Additionally, a moderate positive and significant relationship was found between perceived learning levels in programming and both attitude towards programming (r=0.417, p<0.01) and motivation towards programming (r=0.474, p<0.01). There was also a moderate negative relationship (r=-0.303) that was significant (p<0.01) between perceived learning levels in programming and anxiety towards programming.

Predictive Factors of the Perceived Learning Levels in Programming

The results from the multiple regression analysis predicting perceived learning level in programming, based on variables as attitude towards programming, motivation, anxiety, and self-efficacy perception, are presented in Table 4.

Table 4

Multiple Regression Analysis for Predicting Perceived Learning Level in Programming

Variable	*B*	*SE*	*β*	*t*	*p*	*95% CI for B*	
						LB	*UB*
Constant	0.752	0.400		1,878	0.062	-0.038	1.541
Attitude	-0.050	0.095	-0.038	-0.529	0.597	-0.239	0.138
Motivation	0.177	0.065	0.188	2.742	0.007*	0.050	0.304
Self-Efficacy	0.682	0.071	0.626	9.649	0.000**	0.543	0.821
Anxiety	-0.012	0.047	-0.016	-0.260	0.795	-0.105	0.081

$R = 0.720$ $R^2 = 0.518$ $\Delta R^2 = 0.508$

$F_{(4,192)} = 51.674$ $p < 0.001$

Note: SE: standard error; ΔR^2: $R^2_{adjusted}$; CI: confidence interval; LB: lower bound; UB: upper bound; *p<.01; **p<.001

Table 4 indicates that attitudes toward programming, motivation, anxiety, and self-efficacy significantly predict perceived learning levels in programming ($F(4,192)=51.674$; $R=0.720$; $R^2=0.518$; $\Delta R^2=0.508$; $p<0.001$). These variables explain approximately 51% of the total variance in perceived learning in programming. The standardized regression coefficients (β) reveal that self-efficacy (β=0.626), motivation (β=0.188), attitude (β=-0.038), and anxiety (β=-0.016) are the key predictors of perceived learning in programming. Further examination of the significance of these regression coefficients shows that self-efficacy ($p<0.001$) and motivation ($p<0.01$) are significant predictors of perceived learning in programming, while attitude and anxiety are not.

The multiple regression analysis results provide a regression equation for predicting perceived learning in programming based on the variables of attitude towards programming, motivation, self-efficacy, and anxiety. The equation is as follows:

Perceived learning in programming= 0.752-(0.05×Attitude) + (0.177×Motivation) + (0.682×Self-efficacy) - (0.012×Anxiety)

Discussion

Discussion of Descriptive Results

Students' positive attitudes toward programming may stem from greater exposure to programming environments and an increasing awareness of programming's relevance to their academic and professional lives. This aligns with previous studies that report favorable attitudes (Erol & Kurt, 2017; Korkmaz & Altun, 2013; Özyurt & Özyurt, 2015; Özyurt, 2015). However, Karaci (2016) found that university students demonstrated neutral attitudes towards programming. The differences in these findings may be attributed to the various programming courses taken by the students, as well as the methods and techniques used in the course environment.

The high levels of motivation observed among university students in programming may be linked to the use of engaging instructional strategies and supportive learning environments. This observation aligns with the findings of Özonur (2022), who reported high motivation levels. In contrast, Avcı (2022) found lower motivation among students. Previous research suggests that methods such as gamification and visual programming can significantly boost students' motivation for programming (Çilengir & İzmirli, 2023; Erol, 2015). Therefore, instructional practices and learning environments should be designed to enhance students' motivation and align with their learning objectives (Yağcı, 2016b).

University students exhibit strong self-efficacy perceptions in programming competence and self-regulation, yet their self-efficacy levels regarding programming skills remains moderate. This suggests that students feel confident in understanding program structures using a programming editor and have positive beliefs about programming. Conversely, they encounter challenges when solving problems during programming, resulting in a medium level of perceived competence. This result suggests that while students view themselves as cognitively capable in programming, they feel less skilled in its practical application (Gökoğlu, 2022). According to previous literature, Lin (2016) found that university students had high self-efficacy perception in programming. Additionally, Gökoğlu (2022) found that although university students'

self-efficacy perceptions regarding programming were high, their programming skills were relatively lower than other dimensions. Similarly, Akçay and Çoklar (2018) noted that university students displayed moderate programming self-efficacy. While these students did not face significant difficulties with simple programming tasks, their self-efficacy decreased when confronted with more complex challenges. Furthermore, Yağcı (2016a) reported that university students also demonstrated a medium level of self-efficacy. Therefore, it can be inferred that the choice of programming language or the preferred programming editor used in coursework may influence students' self-efficacy in programming.

The moderate level of anxiety observed among university students in programming is consisted with previous studies research (Engin, 2024; Keskinkılıç & Kalelioğlu, 2024). Factors such as the choice of programming language, instructional design, and students' prior experiences may significantly influence their anxiety levels. This finding shows that while students do not experience severe anxiety, they still face challenges that can hinder their learning process. As a result, it can be inferred that variations in anxiety levels may be influenced by factors such as the programming language used in courses, the learning environment, the teaching methods and techniques employed, as well as the students' years of experience in programming.

University students demonstrated a high level of perceived learning in programming, reflecting confidence in both understanding and applying programming concepts. Similarly, Tokumacı (2019) found that students showed high attitudes toward programming, strong self-efficacy and perceived learning levels. Moreover, Rebuta et al. (2022) reported that students perceived the value of learning programming as being at a medium level. Based on this finding, it can be concluded that students evaluate themselves as competent in the knowledge and skills acquired in the programming process.

Discussion of the Relationship between Factors

The correlation analysis revealed a strong positive association between students' perceived learning levels in programming and their self-efficacy perceptions. Additionally, perceived learning moderately and positively correlated with students' motivation and attitudes toward programming. Conversely, a moderate negative correlation was found between perceived learning and programming-related anxiety. Given these significant relationships, multiple regression analysis was conducted to identify which variables predict perceived learning levels in programming. It was determined that the variables in the model significantly predicted students' perceived learning in programming, explaining 51% of the variation. Analysis of the regression coefficients indicated that perceived learning in programming was influenced by self-efficacy and motivation, while attitude towards programming and anxiety did not significantly affect perceived learning.

The results of the study show that students' self-efficacy in programming is the most significant factor influencing their perceived levels of learning. This indicates that students' confidence in their ability to understand and apply programming concepts is crucial in determining their learning outcomes. This finding is consistent with existing literature, including studies by Abdunami et al. (2019), Cigdem (2015), Ramalingam and Wiedenbeck (1998) and Aşkar and Davenport (2009), which demonstrate that self-

efficacy perceptions in programming correlate with overall achievement. Furthermore, Gürer and Tokumacı (2020b) found a positive relationship between perceived learning in programming and self-efficacy perception of programming. They also pointed out that self-efficacy significantly predicts perceived learning in programming. As students gain experience in programming through their courses, their self-efficacy tends to increase, which, in turn, enhances their perceived learning.

Motivation emerged as the second most influential factor affecting students' perceptions of their learning in programming. This conclusion is supported by previous research (Avcı, 2022) and is likely rooted in the critical role of self-efficacy in motivation (Ramalingam & Wiedenbeck, 1998) as well as its causal influence on motivational dynamics (Zimmerman, 2000). Furthermore, the learning environment in programming instruction can significantly impact students' motivation (Calder, 2010). Several studies indicate that incorporating gamification or visual programming applications into instruction enhances students' motivation for programming and improves their overall achievement (Erol, 2015; Fidan, 2016). Consistent with these findings, the current study shows that students who are more motivated in programming tend to report higher levels of perceived learning. This highlights that motivation strengthens students' sense of competence in programming.

A significant positive correlation was found between students' perceived learning in programming and their attitudes toward the subject. However, it was surprising that attitude did not emerge as a significant predictor of perceived learning in this study. In a study by Gürer and Tokumacı (2020b) that examined factors affecting engineering students' success in computer programming, they found that gender, attitude towards programming, and self-efficacy in programming significantly predicted students' perceived learning. Furthermore, Durak and Bulut (2024) found that programming attitude significantly impacts programming performance. These inconsistencies may arise from differences in sample characteristics, such as the students' academic departments or their prior exposure to programming. This suggests that the impact of attitude on perceived learning may be context-dependent and influenced by both individual and environmental factors.

A negative correlation was observed between students' perceived learning in programming and their programming-related anxiety, although anxiety did not significantly predict their perceived learning levels. Supporting this result, Durak and Bulut (2024) found that programming anxiety did not impact programming performance. Conversely, Yildirim and Ozdener (2021) found that programming anxiety related to significant others and success was correlated. Previous research suggests that students' programming anxiety tends to decrease after receiving programming education (Ayersman & Reed, 1995). This indicates that even if students have anxiety during programming, it does not impact their learning levels.

Conclusion

The study reveals that university students exhibit high levels of attitude, motivation, self-efficacy, and perceived learning in programming, while their anxiety levels are moderate. Significant positive relationships exist between students' perceived learning in programming and their attitudes, motivation, and self-efficacy perceptions. Additionally, there is a significant negative relationship between anxiety and perceived

learning. Students' perceived learning levels in programming were influenced by self-efficacy and motivation.

Implications

Given that self-efficacy is crucial for programming success and that it can be influenced by experience (as noted by Ramalingam et al. 2004), it is essential for instructors to design lesson plans that encourage more hands-on practice. This approach will help students gain experience and, consequently, boost their confidence in their programming abilities. Since motivation is crucial for success in programming, instructors can design more engaging learning environments that encourage active student participation and incorporate various programming applications. For example, incorporating Artificial Intelligence (AI) assisted programming instruction can enhance students' motivation and achievement while reducing their anxiety (Fan et al., 2025). AI-supported programming instruction may therefore serve as an effective approach in programming education. Therefore, students should be given more opportunities to practice, decrease programming anxiety, and improve their performance and attitudes towards programming. Implementing project-based or problem-based learning approaches can contribute to achieving these goals. Additionally, support from instructors or peers can help them with challenges in the programming learning process. Therefore, creating supportive learning environments and offering step-by-step guidance may help reduce programming anxiety and foster more effective learning experiences. In this regard, using collaborative learning or collaborative problem solving approaches for programming instruction can be beneficial.

Limitations

This research was conducted with undergraduate and associate degree students at a state university. To improve the generalizability of the findings, it is recommended that future studies include a larger sample size and involve different departments across various universities. In future research, employing other sampling techniques is advisable to represent the population better. Additionally, future studies may combine qualitative and quantitative research methods to understand better the factors influencing the perceived learning level of programming. To improve our understanding of university students' perceived learning levels in programming, future research should explore the factors that contribute to the significant 49% variance observed. Addressing this gap could lead to the development of more effective instructional strategies and ultimately enhance learning outcomes. Finally, methods like structural equation modeling and artificial neural networks can be beneficial for exploring complex relationships among factors.

Acknowledgements

No funding or financial support of any kind was received by the authors for the study.

Statement of Responsibility

GU: conceptualization, methodology, formal analysis, investigation, writing-original draft. **EÜ**: conceptualization, writing-review & editing, supervision. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest

There is no conflict of interest.

Author Bios:

Gamze Uslu is an instructor in the Department of Computer Technologies at Selçuk University. She received her master's degree in Internet and Information Technology Management from Afyon Kocatepe University, in Turkey. Her research interests are focused on programming and technology management.

Erhan Unal is an associate professor at the Department of Educational Sciences at Afyon Kocatepe University, Turkey. He received his PhD degree in Computer Education and Instructional Technology from Gazi University, in Turkey. His research interests are instructional design, collaborative technologies and constructivist teaching and learning methods.

References

Abdunabi, R., Hbaci, I., & Ku, H-Y. (2019). Towards enhancing programming self-efficacy perceptions among undergraduate information systems students. *Journal of Information Technology Education: Research, 18,* 185-206. https://doi.org/10.28945/4308

Akçay, A., & Çoklar, A.N. (2018). Investigation of perceived self-efficacy of pre-service information technology and software teachers for programming regarding different variables. *Kastamonu Education Journal, 26*(4), 2163-2176. https://doi.org/10.24106/kefdergi.2904

Akkoyunlu, B., & Kurbanoğlu, S. (2004). A study on teachers' information literacy self-efficacy beliefs. *Hacettepe University Journal of Education*, *27,* 11-20. https://dergipark.org.tr/tr/download/article-file/87821

Altun, A., & Mazman, S. G. (2012). Programlamaya ilişkin öz yeterlilik algısı ölçeğinin Türkçe formunun güvenirlik ve geçerlik çalışması. *Journal of Measurement and Evaluation in Education and Psychology, 3*(2), 297-308. https://dergipark.org.tr/en/download/article-file/65965

Amnouychokanant, V., Boonlue, S., Chuathong, S., & Thamwipat, K. (2021). A study of first-year students' attitudes toward programming in the innovation in educational technology course. *Education Research International, 2021*(1), 9105342. https://doi.org/10.1155/2021/9105342

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). https://eric.ed.gov/?id=ED503900

Avcı, Ü. (2022). A predictive analysis of learning motivation and reflective thinking skills on computer programming achievement. *Computer Applications in Engineering Education*, *30*(4), 1102-1116. https://doi.org/10.1002/cae.22505

Avcı, Ü., & Ersoy, H. (2018). The adaptation of learning motivation in computer programming courses scale into Turkish: The study of validity and reliability. *Journal of Higher Education and Science*, *8*(1), 73-81. https://dergipark.org.tr/tr/download/article-file/1711712

Ayalew, Y., Tshukudu, E., & Lefoanea, M. (2018). Factors affecting programming performance of first year students at a University in Botswana. *African Journal of Research in Mathematics, Science and Technology Education*, *22*(3), 363-373. https://doi.org/10.1080/18117295.2018.1540169

Ayersman, D. J., & Michael Reed, W. (1995). Effects of learning styles, programming, and gender on computer anxiety. *Journal of Research on Computing in Education, 28*(2), 148-161. https://doi.org/10.1080/08886504.1995.10782157

Bacon, D. R. (2016). Reporting actual and perceived student learning in education research. *Journal of Marketing Education*, *38*(1), 3-6. https://doi.org/10.1177/0273475316636732

Bandura, A. (1994). Self-efficacy. In V. S. *Ramachaudran (Ed.), Encyclopedia of human behavior* (Vol. 4, pp. 71-81). Academic Press.

Başer, M. (2013a). Attitude, gender and achievement in computer programming. *Middle-East Journal of Scientific Research, 14*(2), 248-255. https://doi.org/10.5829/idosi.mejsr.2013.14.2.2007

Başer, M. (2013b). Developing attitude scale toward computer programming. *The Journal of Academic Social Science Studies*, *6*(6), 199-215. http://dx.doi.org/10.9761/JASSS1702

Baştemur-Kaya, C. (2018). *Effect of use of Alice software on students' academic achievement, problem solving skill perceptions, motivations and readiness level to programming in computer programming teaching* [Unpublished doctoral dissertation]. Gazi University.

Batista, I. V., & Cornachione, E. B. (2005). Learning styles influences on satisfaction and perceived learning: Analysis of an online business game. *Developments in Business Simulation and Experiential Learning*, *32*, 22-30. http://absel-ojs-ttu.tdl.org/absel/article/view/552

Bergin, S., & Reilly, R. (2005). Programming: Factors that influence success. *ACM SIGCSE Bulletin, 37*(1), 411-415. https://doi.org/10.1145/1047124.1047480

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016). Developing computational thinking in compulsory education. *European Commission, JRC Science for Policy Report*, *68*. https://publications.jrc.ec.europa.eu/repository/handle/JRC104188

Breckler, S. J. (1984). Empirical validation of affect, behavior, and cognition as distinct components of attitude. *Journal of Personality and Social Psychology*, *47*(6), 1191-1205. https://doi.org/10.1037//0022-3514.47.6.1191

Büyüköztürk, Ş. (2024). *Sosyal bilimler için veri analizi el kitabı: İstatistik araştırma deseni SPSS uygulamaları ve yorum* (31st ed.). Pegem Akademi Yayınları.

Calder, N. (2010). Using Scratch: An integrated problem-solving approach to mathematical thinking. *Australian Primary Mathematics Classroom*, *15*(4), 9-14. https://files.eric.ed.gov/fulltext/EJ906680.pdf

Caspi, A., & Blau, I. (2008). Social presence in online discussion groups: Testing three conceptions and their relations to perceived learning. *Social Psychology of Education*, *11*(3), 323-346. https://doi.org/10.1007/s11218-008-9054-2

Çetin, İ., & Özden, M. Y. (2015). Development of computer programming attitude scale for university students. *Computer Applications in Engineering Education*, *23*(5), 667-672. https://doi.org/10.1002/Cae.21639

Cigdem, H. (2015). How does self-regulation affect computer-programming achievement in a blended context?. *Contemporary Educational Technology*, *6*(1), 19-37. https://doi.org/10.30935/cedtech/6137

Çilengir, M. D., & İzmirli, S. (2023). The impact of using gamification approach in block-based programming instruction on achievement and motivation. *International Journal of Computers in Education*, *6*(2), 79-103. https://doi.org/10.5281/ZENODO.10447397

Compeau, D. R., & Higgins, C. A. (1995). Computer self-efficacy: Development of a measure and initial test. *MIS Quarterly*, *19*(2), 189-211. https://doi.org/10.2307/249688

Connolly, C., Murphy, E., & Moore, S. (2007, September 3-7). Second chance learners, supporting adults learning computer programming. In *International Conference on Engineering Education–ICEE*. Coimbra, Portugal. http://icee2007.dei.uc.pt/proceedings/papers/407.pdf

Computer Science Teachers Association (CSTA) & International Society for Technology in Education (ISTE) (2011). *Computational thinking leadership toolkit* (1st ed.). CSTA & ISTE. https://cdn.iste.org/www-root/2020-10/ISTE_CT_Leadership_Toolkit_booklet.pdf

Creswell, J. W. (2012). *Educational research: Planning, conducting, and evaluating quantitative and qualitative research* (4th ed.). Pearson.

Demir, F. (2015). *The effect of different usage of the educational programming language in programming education on the programming anxiety and achievement* [Unpublished doctoral dissertation]. Ataturk University.

Demirer, V., & Sak, N. (2016). Programming education and new approaches around the world and in Turkey. *Journal of Theory and Practice in Education*, *12*(3), 521-546. https://dergipark.org.tr/en/download/article-file/262355

Durak, A., & Bulut, V. (2024). Predicting low and high student performance in programming education using PLS-SEM algorithms. *Technology, Knowledge and Learning*, *30*, 1231-1248. https://doi.org/10.1007/s10758-024-09737-2

Engin, M. (2024). Investigating the relationship between individual innovativeness and programming anxiety. *Journal of University Research*, *7*(2), 150-159. https://doi.org/10.32329/uad.1432414

Erol, O. (2015). *The effects of teaching programming with Scratch on pre-service information technology teachers' motivation and achievement* [Unpublished doctoral dissertation]. Anadolu University.

Erol, O., & Kurt, A. A. (2017). Investigation of CEIT students' attitudes towards programming. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi, 1*(41), 314-325. https://doi.org/10.21764/efd.64721

European Schoolnet (2015). *Computing our future: Computer programming and coding - Priorities, school curricula and initiatives across Europe*

http://www.eun.org/documents/411753/817341/Computing+our+future_final_2015.pdf/d3780a64-1081-4488-8549-6033200e3c03

Eurydice (2022). *Informatics education at school in Europe.* Publications Office of the European Union. https://data.europa.eu/doi/10.2797/268406

Fan, G., Liu, D., Zhang, R., & Pan, L. (2025). The impact of AI-assisted pair programming on student motivation, programming anxiety, collaborative learning, and programming performance: A comparative study with traditional pair programming and individual approaches. *International Journal of STEM Education, 12,* Article 16. https://doi.org/10.1186/s40594-025-00537-3

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. https://doi.org/10.1016/j.compedu.2012.11.016

Fidan, A. (2016). *Effect of gamification in teaching programming with scratch on student engagement* [Unpublished master's thesis]. Uludağ University.

Fraenkel, J., Wallen, N., & Hyun, H. (2012). *How to design and evaluate research in education.* McGraw-Hill Education.

George, D., & Mallery, P. (2010). *SPSS for Windows step by step: a simple guide and reference.* Allyn & Bacon.

Gökoğlu, S. (2022). Computer programming self-efficacy scale for computer literacy education: Turkish validity and reliability study. *Bolu Abant Izzet Baysal University Journal of Faculty of Education, 22*(2), 529-551. https://dx.doi.org/10.17240/aibuefd.2022..-654547

Gomes, A., & Mendes, A. J. (2007, September 3-7). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE*. Coimbra, Portugal. http://icee2007.dei.uc.pt/proceedings/papers/411.pdf

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. https://doi.org/10.30935/cet.471004

Gürbüztürk, O., & Tanataş, D. Y. (2024). The effect of block-based coding tools on academic achievement, attitude and computational thinking skill: Meta-analysis study. *Inonu University Journal of the Graduate School of Education, 11*(21), 58-79. https://doi.org/10.29129/inujgse.1425193

Gürer, M. D., & Tokumacı, S. (2020a). Engineering students' attitudes towards programming. *Cumhuriyet International Journal of Education, 9*(4), 1064-1082. http://dx.doi.org/10.30703/cije.671244

Gürer, M. D., & Tokumacı, S. (2020b). Factors affecting engineering students' achievement in computer programming. *International Journal of Computer Science Education in Schools, 3*(4), 23–34. https://doi.org/10.21585/ijcses.v3i4.74

Gürer, M. D., Çetin, İ., & Top, E. (2019). Factors affecting students' attitudes toward computer programming. *Informatics in Education, 18*(2), 281-296. https://doi.org/10.15388/infedu.2019.13

Hongwarittorrn, N., & Krairit, D. (2010, April). Effects of program visualization (jeliot3) on students' performance and attitudes towards java programming. In *The*

spring 8th international conference on computing, communication and control technologies. Orlando, Florida.

Horzum, M. B., & Çakır, Ö. (2009). The validity and reliability study of the Turkish version of the online technologies self-efficacy scale. *Educational Sciences: Theory and Practice, 9*(3), 1343-1356. https://files.eric.ed.gov/fulltext/EJ858927.pdf

Horzum, M. B., Demir Kaymak, Z. & Canan Güngören, Ö. (2015). Structural equation modeling towards online learning readiness, academic motivations and perceived learning. *Educational Sciences: Theory & Practice, 15*(3), 759-770. https://files.eric.ed.gov/fulltext/EJ1067438.pdf

ISTE. (2016). ISTE standards: Students. *International Society for Technology in Education.* Retrieved from https://www.iste.org/standards/iste-standards-for-students

Kanaparan, G., Cullen, R., & Mason, D. (2017). Effect of self-efficacy and emotional engagement on introductory programming students. *Australasian conference on information systems (ACIS).* Hobart, Australia. https://pdfs.semanticscholar.org/305b/845507ade4a9b412840d2acb3a8acb6c8cf6.pdf

Karaci, A. (2016). Investigation of attitudes towards computer programming in terms of various variables. *International Journal of Programming Languages and Applications*, *6*(1/2), 1-9. https://doi.org/10.5121/ijpla.2016.6201

Keskinkılıç, F., & Kalelioğlu, F. (2024). An analysis of programming anxiety levels of associate degree computer department students in terms of various variables. *Journal of Uludag University Faculty of Education, 37*(3), 1092-1110. https://doi.org/10.19171/uefad.1493874

Korkmaz, Ö., & Altun, H. (2013). Engineering and ceit student's attitude towards learning computer programming. *The Journal of Academic Social Science Studies, 6*(2), 1169-1185. https://doi.org/10.9761/jasss_690

Krpan, D., Mladenović, S., & Rosić, M. (2015). Undergraduate programming courses, students' perception and success. *Procedia-Social and Behavioral Sciences, 174*, 3868-3872. https://doi.org/10.1016/j.sbspro.2015.01.1126

Lin, G.-Y. (2016). Self-efficacy beliefs and their sources in undergraduate computing disciplines: An examination of gender and persistence. *Journal of Educational Computing Research, 53*(4), 540-561. https://doi.org/10.1177/0735633115608440

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. https://doi.org/10.1016/j.chb.2014.09.012

Mahmud, M. M., & Wong, S. F. (2022). Digital age: The importance of 21st century skills among the undergraduates. *Frontiers in Education, 7*, 950553. https://doi.org/10.3389/feduc.2022.950553

Mannila, L., Peltomäki, M., & Salakoski, T. (2006). What about a simple language? Analyzing the difficulties in learning to program. *Computer Science Education, 16*(3), 211-227. https://doi.org/10.1080/08993400600912384

Mayer, D. P. (2008). *Overcoming school anxiety: How to help your child deal with separation, tests, homework, bullies, math phobia, and other worries*. Amacom.

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. https://dergipark.org.tr/tr/download/article-file/231311

Mills, K. A., Cope, J., Scholes, L., & Rowe, L. (2024). Coding and computational thinking across the curriculum: A review of educational outcomes. *Review of Educational Research*, *95*(3), 581-618. Shttps://doi.org/10.3102/00346543241241327

Mutanga, M. B. (2020). The effect of cognitive factors in determining students' success in computer programming. *Journal of Theoretical and Applied Information Technology*, *98*(17), 3606-3618. https://www.jatit.org/volumes/Vol98No17/16Vol98No17.pdf

OECD (2024), *OECD Digital Economy Outlook 2024 (Volume 1): Embracing the Technology Frontier*. OECD Publishing. https://doi.org/10.1787/a1689dc5-en.

Olelewe, C. J., & Agomuo, E. E. (2016). Effects of B-learning and F2F learning environments on students' achievement in QBASIC programming. *Computers & Education*, *103*, 76-86. https://doi.org/10.1016/j.compedu.2016.09.012

Olipas, C. N. P., Leona, R. F., Villegas, A. C. A., Cunanan Jr, A. I., & Javate, C. L. P. (2021). The academic performance and the computer programming anxiety of BSIT students: A basis for instructional strategy improvement. *International Journal of Advanced Engineering, Management and Science, 7*(6), 125-129. https://doi.org/10.22161/ijaems.76.15

Özmen, B., & Altun, A. (2014). Undergraduate students' experiences in programming: Difficulties and obstacles. *Turkish Online Journal of Qualitative Inquiry*, *5*(3), 9-27. https://doi.org/10.17569/tojqi.20328

Özonur, M. (2022). Computer programming students' learning motivation in programming courses. *Journal of Advanced Education Studies, 4*(1), 61-69. https://doi.org/10.48166/ejaes.1123170

Özyurt, Ö. (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. https://doi.org/10.17718/tojde.58767

Özyurt, Ö., & Özyurt, H. (2015). A study for determining computer programming students' attitudes towards programming and their programming self-efficacy. *Journal of Theory and Practice in Education, 11*(1), 51-67. https://doi.org/10.17244/eku.53204

Pintrich, P.R., & Schunk, D. H. (1996). *Motivation in education: Theory, research, and applications*. Merrill.

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. https://doi.org/10.2190/C670-Y3C8-LTJ1-CT3P

Ramalingam, V., LaBelle, D., & Wiedenbeck, S. (2004, June 28-30). Self-efficacy and mental models in learning to program. In *Proceedings of the 9th Annual SIGCSE*

Conference on Innovation and technology in Computer Science Education (pp. 171-175). Leeds, United Kingdom. https://doi.org/10.1145/1007996.1008042

Rebuta, K. M. N., Cabaron, I. M. P., Pucong, R. J. C., Bisquera, J. M. C., Llerado, R. T., & Buladaco, M. V. M. (2022). Relationship of programming skills and perceived value of learning programming among information technology education students in Davao del Sur. *International Journal of Research and Innovation in Social Science (IJRISS)*, *6*(6), 882-887. https://doi.org/10.47772/IJRISS.2022.6633

Reich-Stiebert, N., Eyssel, F., & Hohnemann, C. (2019). Involve the user! Changing attitudes toward robots by user participation in a robot prototyping process. *Computers in Human Behavior*, *91*, 290-296. https://doi.org/10.1016/j.chb.2018.09.041

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, *13*(2), 137-172. https://doi.org/10.1076/csed.13.2.137.14200

Rogerson, C., & Scott, E. (2010). The fear factor: How it affects students learning to program in a tertiary environment. *Journal of Information Technology Education: Research*, *9*(1), 147-171. https://doi.org/10.28945/1183

Rountree, N., Rountree, J., Robins, A., & Hannah, R. (2004). Interacting factors that predict success and failure in a CS1 course. *ACM SIGCSE Bulletin*, *36*(4), 101-104. https://doi.org/10.1145/1041624.1041669

Santos, S. C., Tedesco, P. A., Borba, M., & Brito, M. (2020). Innovative approaches in teaching programming: A systematic literature review. In *Proceedings of the 12th International Conference on Computer Supported Education* (Vol. 1, pp. 205-214). https://doi.org/10.5220/0009190502050214.

Schunk, D. H. (2012). *Learning Theories an Educational Perspective*. Pearson.

Şişman, B., & Küçük, S. (2018). Pre-service teachers' flow, anxiety and cognitive load levels in robotics programming. *Educational Technology Theory and Practice*, *8(*2), 108-124. https://doi.org/10.17943/etku.366193

Tokumacı, S. (2019). *Factors predicting engineering faculty students' perceived learning of computer programming* [Unpublished master's thesis]. Bolu Abant İzzet Baysal University.

Vitasari, P., Wahab, M. N. A., Othman, A., Herawan, T., & Sinnadurai, S. K. (2010). The relationship between study anxiety and academic performance among engineering students. *Procedia-Social and Behavioral Sciences*, *8*, 490-497. https://doi.org/10.1016/j.sbspro.2010.12.067

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

Yağcı, M. (2016b). Blended learning experience in a programming language course and the effect of the thinking styles of the students on success and motivation. *The Turkish Online Journal of Educational Technology*, *15*(4), 32-45. https://files.eric.ed.gov/fulltext/EJ1117633.pdf

Yıldırım, O. G. (2022). *An action research study on the development of object-oriented programming course* [Unpublished doctoral dissertation]. Marmara University.

Yildirim, O. G., & Ozdener, N. (2021). An action research study on the development of object-oriented programming *course. International Journal of Technology in Education and Science (IJTES), 5*(4), 620-628. https://doi.org/10.46328/ijtes.277

Yıldırım, O. G., & Özdener, N. (2022). The development and validation of the programming anxiety scale. *International Journal of Computer Science Education in Schools, 5*(3), 1-18. https://doi.org/10.21585/ijcses.v5i3.140

Yıldız, T., & Seferoğlu, S. S. (2021). The effect of robotic programming on coding attitude and computational thinking skills toward self-efficacy perception. *Journal of Learning and Teaching in Digital Age, 6*(2), 101-116. https://files.eric.ed.gov/fulltext/EJ1308356.pdf

Yılmaz, F., & Çakır, H. (2019). Investigation of factors affecting two-year vocational college students' computer programming achievement. *Educational Technology Theory and Practice*, *9*(2), 416-437. https://doi.org/10.17943/etku.527202

Yılmaz, R., & Yılmaz, F. G. K. (2023). The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. *Computers and Education: Artificial Intelligence*, *4*, 100147. https://doi.org/10.1016/j.caeai.2023.100147

Zimmerman, B. J. (2000). Self-efficacy: An essential motive to learn. *Contemporary Educational Psychology, 25*(1), 82-91. https://doi.org/10.1006/ceps.1999.1016

This is an Open Access article distributed under the terms of the Creative CommonsAttribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0). For further information, you can refer to https://creativecommons.org/licenses/by-nc-sa/4.0/