Computer Programming Self-Efficacy Levels of Engineering Bachelor Students at Eastern Mediterranean University (EMU)

Golmehr Esmaeilzadeh Hanjani

Submitted to the
Institute of Graduate Studies and Research
in partial fulfillment of the requirements for the Degree of

Master of Science
in
Information and Communication Technologies in Education

Eastern Mediterranean University
February – 2019
Gazimağusa, North Cyprus

Approval of the Graduate Committee of School of Computing and Technology

Assoc. Prof. Dr. Ali Hakan Ulusoy
Acting Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science in Information and Communication Technologies in Education.

Assoc. Prof. Dr. Ersun İşçioğlu
Acting Chair, Department of Computer Education and Instructional Technologies

We certify that we have read this thesis and that in our opinion it is fully adequate in scope and quality as a thesis for the degree of Master of Science in Information and Communication Technologies in Education.

Assoc. Prof. Dr. Ersun İşçioğlu
Supervisor

Examining Committee

1. Assoc. Prof. Dr. Ersun İşçioğlu
2. Asst. Prof. Dr. Fahme Dabaj
3. Asst. Prof. Dr. Fatma Tansu Hocanin

ABSTRACT

This study assessed Computer Programming Self-Efficacy (CPSE) Levels of engineering bachelor students at Eastern Mediterranean University (EMU). The research method of this study was quantitative method, using the survey technique furthermore a questionnaire with two sections consisting of demographic information plus a computer programming self-efficacy scale, was distributed among 510 engineering students from five different engineering departments, including Computer Engineering, Electrical & Electronics Engineering, Mechanical Engineering, Industrial Engineering and Civil Engineering departments.

According to the results the CPSE level of engineering students at EMU can be measured as an average level also in some computer programming related tasks female have a higher CPSE level than male, on the other hand by comparing 5 engineering departments, it can be said that Computer Engineering and Electrical-Electronics Engineering students have a higher CPSE level than others however Civil Engineering students had the lowest CPSE level. Additionally, there was no noteworthy difference among students according to their grade (class level) but overall the CPSE level of 4^{th} Year students were higher than others.

Keywords: Self-Efficacy, Computer Programming Self-Efficacy, Engineering Students

ÖZ

Bu çalışma, Doğu Akdeniz Üniversitesi'ndeki (DAÜ) Mühendislik lisans öğrencilerinin Bilgisayar Programlama Öz-Yeterlilik (CPSE) Düzeylerini incelemektedir. Çalışma nicel araştırma olarak planlanmış ve tarama metodu kullanılmıştır. Veri toplama aracı demografik bilgiler ve bilgisayar programlama öz-yeterlilik ölçeği olmak üzere iki bölümden oluşturulmuştur. Çalışmaya, bilgisayar mühendisliği, elektrik-elektronik mühendisliği, makina mühendisliği endüstri mühendisliği ve inşaat dahil olmak üzere beş farklı mühendislik bölümünden 510 mühendislik öğrencisi katılmıştır.

Elde edilen sonuçlara göre, mühendislik öğrencilerinin bilgisayar programlama öz-yeterlik seviyelerinin orta düzeyde olduğu belirlenmiştir. Diğer taraftan cinsiyet açısından sonuçlar incelendiğinde, tüm bölümlerde kadınların CPSE düzeyinin erkeklerden daha yüksek olduğu söylenebilir. Ek olarak, Bilgisayar Mühendisliği ve Elektrik-Elektronik Mühendisliği öğrencilerinin CPSE seviyelerinin diğer bölümlerdeki öğrencilerden daha yüksek olduğu, İnşaat Mühendisliği öğrencilerinin de en düşük CPSE seviyesine sahip olduğu ortaya çıkmıştır. Ayrıca, tüm bölümlerde 4. Sınıf öğrencilerinin CPSE düzeyinin diğer sınıflardakilere göre daha yüksek olduğu belirlenmiştir.

Anahtar Kelimeler: Öz-Yeterlik, Bilgisayar Programlama Öz-Yeterliği, Mühendislik Öğrencileri

TABLE OF CONTENTS

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 Introduction

People recognize that they should be knowledgeable about computers, if their purpose is to be productive in society (Levy & Murnane, 2004) and they look for answers for they key concerns due to the present advancements in data innovations and technology. According to Hughes (2004), most people consider technology as using computers and the internet but the definition of technology is usually consisting of numerous examples involving utilizing transportation, communication, energy and etc. What was known as “practical arts”, “applied science” and “engineering” before the Second World War and the 20th century; is called technology in our time (Hughes, 2004). With the aim of benefiting from technology, societies should know and learn about its complex and numerous features. Usually, people’s understanding of technology is not complete and enough. Some have an incomplete idea of technology, thinking that technology is restricted to electrically powered or electronic objects. Technology has a huge and significant effect on environment and society; therefore, it cannot be ignored. Societies should have a nuanced comprehension of technology if they want to decide about its advancement, usage and instruction (Lachapelle, Cunningham, & Oh, 2018). Technology is confusing, challenging to describe and complicated. Yet these days most people disregard technology's complexity and its conflicts (Hughes, 2004).

With the help of technology people are able to do a task that would take quite a while completing, in a short period of time, in addition, society expects youngsters growing up within technology age, to utilize technology, as well as to create new technological devices and applications (Kalelioglu, 2015). With the purpose of doing so, computational thinking skills of children should be advanced. Computational Thinking is stated as a talent that everyone, particularly computer scientists should possess (Wing, 2006). It is believed that one of the essential life skills required in the 21st century is the ability of computational thinking and programming, exposes students to computational thinking (Wing, 2006). Therefore, in the USA computational thinking has gained traction for the K-12 schools context since 2006. Many characteristics of 21st century competencies such as creativity, critical thinking, and problem-solving are matched with computational thinking (Binkley et al., 2014). Computational thinking has an important part in education. After internet was widespread people started using it in different aspects such as to get in touch with others, learning, researching, reading and buying books (Borgman et al., 2008), also computational thinking gradually had a role in peoples life. Although Berland & Lee (2011), assumed that individuals can work on one's computational thinking instinctively using non-traditional and non-computational instruments such as tactical board games; according to what educators (Kafai & Burke, 2013) asserted computational thinking is taught in schools via computer programming therefore, programming is vital for K-12 students in this era but simple introduction to technology and computers does not assure people being without stress while working with these devices.

One of the foundations of problem solving for technical and non-technical students, is programming skills (Hilal, Suzastri, & Zulhazlin, 2018). There are some traditional computer programming languages which are similar to the computer's way of thinking like Java or C++, then again some visual programming languages use symbols more like human language and their commands are similar to spoken English thus to simplify computational thinking in K-12 contexts, using visual programming languages rather than traditional programming languages is recommended. Students are allowed to concentrate on the logic and structures of programming plus acquiring computational thinking skills more easily by using visual programming languages; in which case it's not required to study complicated programming syntaxes (Kelleher & Pausch, 2005; Smith, Cypher, & Tesler, 2000).

Learning programming may be hard for novices irrespective of their age. Along with the difficulties of acquiring how to figure structured results for problems and recognizing how programs perform, new programmers also need to study rigid syntax and commands, which might be devastating and sometimes disappointing for elementary programmers (Kelleher & Pausch, 2005). So a question that comes to mind is that why novices should know programming but there are several reasons and motivations to learn how to program including pursuing programming as an occupation pathway or learning problem solving by using a structured method or encoding software personalized systems for individual use (Kelleher & Pausch, 2005). Gomes & Mendes (2007), state that there are difficulties related to acquiring computer programming skills in terms of academic accomplishment and level of learner's satisfaction. Engagement is when students aggressively contribute in the learning procedure. Student engagement is related to their academic success. It is proved that high levels of self-efficacy is linked to high levels of engagement in the classroom (Bilge, Tuzgöl Dost, & Çetin, 2014). Students burnout and school engagement have key variables like study habits and self-efficacy beliefs in other words students should have self-efficacy beliefs in order to have a good performance in school. According to the researches concerned with school engagement, students who have progressive study habits and high self-efficacy beliefs also have high school engagement2 (Bilge et al., 2014).

Self-efficacy is particularly essential and possibly valuable with respects to education and that is because there are some hypothesis stating that people's genuine performance effects their self-efficacy therefore it can impact the upcoming executions. Unlike people with low self-efficacy, individuals with high self-efficacy are bound to carry out challenging responsibilities and progressively endeavor to finish them. Additionally, people with low self-efficacy, may encounter stress and hopelessness and have less motives to do the tasks because they have a narrow vision (Askar & Davenport, 2009).

Fang (2012); Garner (2009) and Nilsen & Larsen (2011) commonly claim that as a result of low self-efficacy and motivation, students experience problems with programming courses and they do not comprehend the main content and algorithmic structure of the programming.

For universities and high schools teaching programming has a significant role in engineering related courses (Dagiene, Skupas, & Kurilovas, 2014). There are many approaches for software engineering so it is required to have a good understanding and enough information about various methods and tools to be a software engineer (Bell, 2005). Teachers know how important it is for students to adopt a useful mental model of programming also, a strong and precise mental model influences course performance and similarly intensifies self-efficacy which is one of the crucial factors in curriculum accomplishment (Ramalingam, LaBelle, & Wiedenbeck, 2004). Programming courses have impact on the improvement of thinking and as a result students comprehend mechanical function of computers and establish theoretical perception of information technologies (Dagiene et al., 2014). The programming courses in universities should be practical so that the students would be ready for coding in a world that computers are significantly important, however lessons about programming languages are for people who want to be successful in a practical way not just in academic studies (Gencturk & Korucu, 2017).

A study was conducted on the essentials of computer programming and the researchers (Brito & De-Sá-Soares, 2014), specified the obligation of adding to the occurrence of evaluations because they believed providing consistent feedback on learners enactment would cause less drop out, since if students know they cannot solve a programming problem earlier they would have time to recover (Brito & De-Sá-Soares, 2014). Engineering education usually changes according to the need and requirement of societies, effective change plans are the ones which engage the faculty with industry expertise and also changes should be essential and prevalent to be unfailing (Graham, 2012). Furthermore here was a research done by Li, Ko, & Zhu, (2015) that showed graduates need programming skills along with being capable to operate with others.

Advancing students' perception of programming ethics is one of the principal objectives of teaching computer programming which as its result, students will be able to practice on explaining programming exercises. On the other hand, programming increases students level of creativity (Dagiene et al., 2014). On the other hand, language programs change as well as people needs and computer operating systems (Bell, 2005).

For example, in Federal University of ABC, for bachelor's degree of Science and Technology, a course is offered as an introduction to programming which has both laboratory and lecture sessions and through that the students will learn about introduction to algorithms, variables and data types, debugging and etc. At first students will learn a programming language which is close to their mother tongue and is called pseudo- programming language, afterwards Java programming language and then Python programming language are presented to them (Zampirolli, Goya, Pimentel, & Kobayashi, 2018).

Scholars believe that instructors would have a better vision if they care about student's opinions of their own capabilities in addition to student's proficiency(Govender et al., 2014). Attitude and self-efficacy perception of learners are the most important aspects of factors affecting the accomplishment in the learning procedure (Anastasiadou & Karakos, 2011; Erdogan, Aydin, & Kabaca, 2008). It is reported that self-efficacy has direct effect on the process of gaining new skill and using the learnt skill. Therefore, to guess one's performance the level of self-efficacy is a dependable indicator (Askar & Davenport, 2009).

Self-efficacy is belief in one's capacity to succeed at tasks. That is to say, self-efficacy is an individual's belief in his or her innate ability to achieve goals. It is a personal judgement of how well one can execute courses of action required to deal with prospective situations (Bandura, 1986) in other words self-evaluation of a person.

How long effort will an individual sustain in facing obstacles and whether if she/he is able to exhibit coping behavior is determined by expectations of self-efficacy. It is thought that determining the self-efficacy levels of individuals could be used as a means of increasing their success as it provides feedback about their performance (Askar & Davenport, 2009). Several researchers stated that computer anxiety, computer self-efficacy and emotional excitement are linked (Marakas, Yi & Johnson, 1998). Computer self-efficacy has an inverse relation with computer anxiety. Computer anxiety was investigated via factor analysis by Beckers & Schmidt (2001), and they found six fundamental dimensions, computer self-efficacy included. Computer self-efficacy is one's perceived talent to learn new abilities or finishing computer tasks (Marakas et al., 1998). People support and develop their efficacy in various areas and in different levels (Bandura, 2006). Perceived efficacy is related to capability judging but self-esteem is the belief of self-worth, they are completely dissimilar phenomena (Bandura, 2006).

Individuals can evaluate their efficacy according to their performance in a wide range of areas of activity or in specific fields. Efficacy generality can be different in a variety of activities, situations in which abilities are shown. Evaluations related to different areas of activity and social traits indicate the pattern and general level of an individual's belief in their performance and efficacy. The efficacy beliefs that people base their lives on are the most essential self-beliefs (Bandura, 2006).

As for the growth of social and cognitive skills, efficacy familiarities and personal control are fundamental (Bandura, 1989). People will be satisfied about themselves when they can achieve important objectives and goals, they also are encouraged to strengthen their efforts in that path. For practicing personal control over motivation, there are some cognitive elements like perceived self- efficacy which have prominent and effective roles. People who are not sure about their ability to complete a task, can easily fail but people who are self-assured will try harder to get to a level of success that they want in other words that the achievements and results of small tasks are motivating or discouraging for someone depends on her/his beliefs for achieving the goals they set for themselves (Bandura, 1989).

According to Bandura (1989), the evaluation of personal efficacy is a part of peoples' decisions on every activity they do. People's thinking patterns (self- hindering or self-enhancing) and the amount of stress and hopelessness that they encounter in the prediction of transactions with the environment, are influenced by evaluation about their abilities. The evaluation of self-efficacy is based on four main causes:

1. experiential mastery
2. vicarious experiences to assess abilities and comparing with others
3. verbal mastery
4. physiological conditions according to which individuals judge their abilities, strength, and liability

Self-efficacy has some sources and one of them is mastery experiences, which is about involvement with a task and the fact that someone can make a failure to a success. One other source can be verbal persuasion which is the reaction that people get about how well a task was performed or positive attitude showing that person can do a job (Bandura, 1997). There are also career sell efficacy and academic self-efficacy that are vital elements in live (Norrbom, 2018).

Askar & Davenport, (2009) explored the computer programming self-efficacy of students with regards to the usage of computer by mother, father and siblings and they state that the usage of the computer by the mother of the family has a huge influence on the students computer programming self-efficacy also the students whose siblings used computers had higher self-efficacy but there was no major difference concerning the fathers usage of computer. According to Askar & Davenport, (2009) one of the main areas of self- efficacy in educational environments is to discover its relevance to the choice of subject/ profession. The results of their study confirmed this relation because they found out that computer engineering students have higher self-efficacy in computer programming comparing to students in electronics or industrial engineering.

Gökçearslan, Günbatar & Kukul (2017), established a Computer Programming Self-Efficacy Scale for secondary school students according to the 32 item computer programming self-efficacy scale of Ramalingam and Wiedenbeck in 1998 because they believed there was a need for an evaluation instrument advanced precisely for secondary school students since computer programming became a predominant course in K12 schools. This study had 233 students as participants who were 12,13 and 14 years old besides they have taken programming courses during IT and software classes. The resulted scale was established throughout statistical procedures on the data the participants gave. In this study 53.6% of the participants were female students and 46.4% were male students so it can be generalized to both genders.

There was another study on secondary students by Yildiz Durak (2018), who had some aims for this study including measuring the students programming self-efficacy to see if digital story design applications in teaching programming has any impact on their self-efficacy. The scale used for this research was the one developed by Gökçearslan et al., (2017) which had 31 items and was appropriate for secondary students. Yildiz Durak (2018), concluded that the students programming self-efficacy didn't have an extremely change throughout the experiment using digital storytelling.

The students who participated in Soykan & Kanbul's (2018) research were 193 participants who took "Coding Education" at a private school via Code.org as an optional course. Code.org is an official website (http://code.org/) founded by twin Iranian brothers and it is for both students and educators to help people gain more knowledge about computer science. This website can be used in 56 languages and it helps individuals to learn the fundamentals of computer programming and computer science through being committed to free curriculum and open source technology. Soykan & Kanbul (2018), believe that learning coding at a young age will help learners to be ready for job opportunities; and that's why they did this research to evaluate the computer programming self-efficacy level of k12 students who took code.org lessons and compare them to students who didn't take coding courses and also students who want to take some coding lessons.

Students usually struggle and commonly think that computer programming courses are challenging (Askar & Davenport, 2009). Also there is a high abrasion for learning programming, but computer programming is one of the essential courses an engineer should pass (Maddrey, 2011). So some students don't have a positive approach towards programming which has a destructive outcome for their success (Korkmaz & Altun, 2014). Therefore, to add to the literature it was desired to perform this study which is focused on engineering bachelor students of Eastern Mediterranean University to verify their level of computer programming self-efficacy.

1.2 Aim of the thesis

The aim of this thesis was to investigate the computer programming self-efficacy level of engineering students at Eastern Mediterranean University and to find an appropriate response for the research questions.

1.3 Research questions

According to the aim of this thesis to investigate the computer programming self-efficacy level of engineering students at Eastern Mediterranean University, there are some research questions which will be answered:

1. What are the computer programming self-efficacy level of engineering students?
2. Is there any significant difference on the computer programming self-efficacy level of engineering students according to gender?
3. Is there any significant difference on the computer programming self-efficacy level of engineering students according to their department?
4. Is there any significant difference on the computer programming self-efficacy level of engineering students according to their class level?

1.4 Significance of the study

According to Bandura (2006), realizing in what way the logic of efficacy is built and in what manner does it work, helps to form experiences that allow people to comprehend desired personal and social differences. And that’s why adding to the computer programming self-efficacy literature is important. With the purpose of evolving new educational approaches to solve the issues related to computer programming, based on previous researches, evaluating self-efficacy is absolutely essential. This study is significant for the university because it is conducted to know more and define computer programming self-efficacy level of bachelor students studying in engineering programs also it is going to be useful for both students and teachers to have a better study plan.

1.5 Limitations

This study was limited to engineering bachelor students who were registered for fall 2018-2019 semester at Eastern Mediterranean University.

1.6 Key terms

Self-efficacy: An individual's belief in his or her capacity to execute behaviors necessary to produce specific performance achievements (Bandura, 1997).

Computer Programming Self-Efficacy: One’s perception and confident in her/his ability to do computer programming related tasks.

Chapter 2

LITERATURE REVIEW

This chapter offers a brief literature review on works that payed attention to computer programming self-efficacy.

2.1 Introduction

People who live in this era are living in the information technology age. Most common devices used every day like smartphones, smart televisions, the Internet, digital cameras are an important part of life in each society. Technology has an extensive and deep influence on different aspects of life (Dao, 2018).

Originally the word technology comes from the Greek word, tekhnē which means ‘art, craft’. As said by Hughes (2004) the use of the word technology started at seventeenth century but it didn’t have the same meaning back then, the usage of this word was in fields like industrial and practical arts. At that time “technology” was defined as the use of science to the practical arts, discoveries and human inventiveness. According to Cambridge dictionary, “technology” the study and knowledge of the pragmatic and industrial, use of scientific findings. Around 1958 some American historians and social scientists gathered and named their group as the society for the history of technology, they chose the word technology instead of engineering because they thought it’s more wide-ranging. Then they started publishing journals with the title of technology and culture. In the first two decades’ rockets, machine tools, telegraph, computers, metal, chemical, land transport and etc. were all referred to as technology in journals pages. Some researchers agree that technology is whatever that can construct and administer a human-built world (Hughes, 2004), So technology is related to mechanics, inventors, engineers, etc.

Technology may even help with education, there are many studies on students notetaking by using a laptop or other devices in the classroom instead of the traditional way of using pens and papers; which some show using technology guides to a better educational result and some conclude that note-taking with technology does not always assist the learning procedure (Kutta-JR, 2017). Additionally, computer engineering is an educational subject which covers computer science and electrical engineering. Computer engineering is described as a field that expresses the science, technology of design, formation, enactment and preservation of software and hardware mechanisms of modern computing systems and computer-controlled devices (IEEEComputerSociety, 2004). Computer engineers are tangled with the design of computer-based systems to deal with particular application needs. They create computers and computer-based systems consisting of hardware and software to explain engineering dilemmas (IEEEComputerSociety, 2004). Then while talking about computer software engineering, computer programming comes to mind which as stated by Cambridge dictionary, it is defined as the activity or job of writing programs for computers.

2.2 Computer programming

Nowadays computers are found everywhere, inside and outside schools plus youth are introducing new things with technology by creating art projects, making video games and etc. (Kafai & Burke, 2013). The usage of computers specially for educational purposes is increasing (Khorrami-Arani, 2001). As for defining computer software and hardware according to (Patzek & Juanes, 2006), software is the simple methods sequenced step by step describing to a computer how to complete a complex task in detailed. On the other hand, hardware is a device which accomplishes those steps. In actual life, people do not describe precisely what they want to say in details because it's not needed since individuals can use body language and other human features to express themselves but the instructions given to a computer must be precise and clear (Patzek & Juanes, 2006). From 1960s afterwards, academics have created several programming languages and settings so that more people could do coding (Kelleher & Pausch, 2005). Wing (2008), thinks of a programming language as a concept of an array of series which can influence some computation when decoded. As humans talk to each other using languages, there is a need for a programming language to talk to a computer and as well as people who have different alphabets and languages there are several computer programming languages (Patzek & Juanes, 2006).

A decent engineer or scientist should understand how a digital computer works and what it is capable of. Someone who wants to continue in the computer business world, should have a reliable and beneficial understanding of primary computational algorithms and coding methods (Patzek & Juanes, 2006). For example according to Jackson & Miller (2009), at Massachusetts Institute of Technology the undergraduate syllabus for computer science and electrical engineering gets restructured when needed and sometimes new courses will be added such as a base course in programming. And by analyzing the performance of the students, it's shown that courses like mathematics, calculus should be the prerequisite for some software engineering courses. So in engineering education and designing a curriculum paying enough attention to theories and practical tools at the same time also knowing that there are several concepts which engineers must be comfortable with; are important.

2.3 Self-efficacy

Efficacy beliefs have impact on people despite the way they think in other words efficacy effects individuals who are purposeful, unpredictable, enthusiastic or even doubtful, furthermore efficacy beliefs influence decision making while people want to do tasks or set an aim or oblige themselves for completing a responsibility/mission; they likely have an impact on the degree and amount of energy a person should spend for an activity, expectations of the results, withstanding the difficulties, flexibility towards danger and etc. (Bandura, 2006). According to some researchers there are several factors about a task that can have a direct influence on the development of perceptions of self-efficacy like how difficult a task may seem to be, the uniqueness of the task or the complexity (Marakas et al., 1998).

While working with a self-efficacy scale it is important to know that if the scale is not focused on the appropriate elements which do not have any influence on the area of self-efficacy, the results won't produce an analytical relation. This is one of the keys on researching in this field because as a result of destructive conclusions, there will be broken theories instead of more accurate data on the literature (Bandura, 2006).

Self-efficacy can be measured in quite a few ways (Pajares, 1997). Self-efficacy scales are one of them which are required to be shaped to evaluate the many-sided efficacy beliefs, these scales should be related to factors which can control the excellence of performance. Perceived efficacy scale is supposed to calculate the levels of anxiety symbolizing the progressions of tolerating disorders for a productive performance.

Self-efficacy evaluations show the level of hardship people are certain they can overcome (Bandura, 2006).

There is a need for primary exertion to detect the structures of obstacles for creating scales to evaluate self-regulatory efficacy and for that by using open-ended interviews and surveys individuals are requested to explain issues that make it difficult for them to do a necessary task on a regular basis. Then factors, issues and related obstacles which are recognized in terms of effectiveness will be constructed into the efficacy items for the scale. Contributors evaluate their ability for facing those issues or overcoming the obstacles. In order to prevent having an upper limit data, adequate levels of difficulties must be considered for the efficacy items (Bandura, 2006). Hence for the standard version of evaluating self-efficacy beliefs, participants will be given items that represent various levels to assess their confidence on trusting their ability to perform the necessary tasks. They should score their efficacy beliefs on for example a 100-point scale, starting at 0 ("cannot be done"); to an average level of certainty showing with 50 ("relatively clear it can be done"); to the guarantee level which is 100 ("absolutely confident") basically it's from zero to a maximum strength, without consisting negative numbers. Scales which use insufficient steps cannot be trusted because they won't be responsive or dependable. Effectiveness scales are unified, ranging from 0 to maximum power and they do not include a negative number (Bandura, 2006).

2.4 Computer programming self-efficacy

Self-Efficacy shows people's perception of themselves according to the past accomplishments and it has a crucial impact on their intentions for upcoming goals. All the definitions and decryptions about self-efficacy have a key point in common which is that self-efficacy is a combination of various factors and those factors have their own influence on one's perception of himself and his performance (Marakas et al., 1998). If societies try to update their knowledge about computer self-efficacy as a result they will have a higher rate of technology acceptance and more practical activities in computer education (Marakas et al., 1998).

There are also some other terms like General Computer Self Efficacy which states people's evaluation of efficacy about various computer application fields, task specific computer self-efficacy which discusses about people's sensitivity of efficacy while doing precise computer-related assignments. That means computer self-efficacy can be assessed in both general and task specific levels that means people's perception of their ability of using specific tools for performing an action can be assessed regardless of the action (Marakas et al., 1998).

Nowadays good educational systems make students ready for a computer dominated world (Khorrami-Arani, 2001). So having computational skills and consequently knowing computer programming is required for this century. On the other hand, if someone has doubt about her/himself or doesn't have enough enthusiasm, despite having the knowledge or abilities required to carry out a certain task, she/he might not succeed. Self-efficacy is an assessment of someone's faith and confidence in their own potential to accomplish in a specific situation. Also it has a dominant impact on what a person does. Besides for students with low computer programming self-efficacy, there is a high chance of being failed in programming courses (Askar & Davenport, 2009). Also as being said by Askar & Davenport (2009) defining the self-efficacy level of people could be helpful as a resource to accumulate their accomplishments considering that it offers feedback regarding peoples performance. One of the values for evaluating a performance is self-efficacy level of the performer. For strengthening student performance, one way is paying attention to self-efficacy, so challenging the students through programming courses does not mean weakening their self-efficacy with difficult programming assignments (Ramalingam et al., 2004).

2.5 Related researches

Murphy, Coover, & Owen, (1989) created a 32-item computer programming self-efficacy scale after going over the literature and checking with five experts. The responses to the scale was designed according to a 5 point Likert scale in which a high mark implied a great level of confidence in someone's judgment of herself/himself while working with computers. A total number of 414 graduate students volunteered for this research so that the scale can be tested from each aspect. After the factor analysis the reliability of the scale was checked. In the end the computer self-efficacy scale had three different levels of computing skills including novice, advanced and mainframe computing skills.

Researching computer self-efficacy is the title of a study by Khorrami-Arani (2001). This study was to assess computer self-efficacy of 8 grade students taking IT lessons. The questionnaire distributed among the students had three parts, the computer self-efficacy scale section was originally developed by Murphy, Coover, & Owen, (1989), but for this study it was modified to be understandable for the students. The results showed that the mean of computer self-efficacy of 105 students was 4.1 out of 5 with a 0.64 standard deviation (Khorrami-Arani, 2001).

Another research in the field of self-efficacy and mental models was by Ramalingam, LaBelle & Wiedenbeck (2004), which examined the relationship among self-efficacy, mental models and computer programming. The instrument used for gathering data in this study was the computer programming self-efficacy scale developed by Ramalingam V. & Wiedenbeck S in 1998 which included thirty-three questions and a 7point Likert scale as answers. Seventy-five bachelor students studying at a public university, participated in this study. The researchers collected information in two stages and they had 2 series of responses to the programming self-efficacy scale, the first one was gathered in the second week of the student's classes and the second one in the thirteenth week while the C++ programming classes were held for fifteen weeks. Comparing the results of these scales, the self-efficacy mean got higher for 68.75, in other words there was a huge increase in the level of computer programming self-efficacy of students.

Askar & Davenport, (2009) did a research on factors related to self-efficacy for Java programming among engineering students. The outcome was that the self-efficacy of males was higher than that of females for Java programming and computer engineering students' self-efficacy scores were considerably higher than that of students from the other engineering departments. In this study on first year engineering and science students the researchers wanted to find the answers for 4 research questions which were about the relationship between gender, department of study, previous computing skills, computer use by family members and self-efficacy for Java programming. The volunteers were from both genders studying in three different engineering departments at Bilkent University, in Ankara, Turkey. As for the results of this study, they came to the point that the male students 'self-efficacy was greater than that of females. By using a regression analysis on the data collected for this study, it was shown that computer familiarity has a linear involvement with computer programming skills which illustrates that the more experience someone has with computers the higher self-efficacy she/he has.

Maddrey, (2011) worked on impacts of problem solving lessons on the programming self-efficacy and achievement of preparatory computer science students. The research experiment required two groups, a control group which included students who attend their regular classes and an experimental group who took part in an online course about non-mathematical problem solving skills. The data analysis showed no considerable alteration in the groups self-efficacy level.

Another research (Davidsson, Larzon, & Ljunggren, 2013) was done to see if there will be a change in programming self-efficacy level of students from the preparatory course up to one year later. The contributors for this study were 77students at Uppsala University, who some were taking an introductory programming course at the time of research and some had already taken that lesson a year before. They used the thirty-two computer programming self-efficacy scale and the results displayed that according to the questions in some skills, programming self-efficacy was reduced and some improved after one year; but the overall programming self-efficacy of students stayed the same over one year (Davidsson et al., 2013).

In 2014 a research was conducted by Korkmaz & Altun (2014), about self-efficacy perceptions of electrical & electronics engineering and computer engineering students. Both quantitative (scale) and qualitative methods were used in this study. For data collection a thirty-two item computer programming self-efficacy scale was used. The participants were 378 junior and senior students from 4 different universities who have passed C++ programming courses. The results showed that self-efficacy level of computer engineering students were higher than that of students who were studying in electrics & electronics however Korkmaz & Altun (2014), think that this difference is natural due to the number of computer programming courses students take in each of these departments.

Govender et al. (2014), carried out a research on the problem solving education benefits. In some geographical regions people suffer from lack of knowledge, technology etc. To help with the practical skills and educational subjects, a support program for some IT teachers in South Africa was put into action. Volunteers were 96 students from 6 secondary schools and six teachers. The methodology used was a qualitative approach by the help of questionnaires and interviews. The questionnaire distributed among participants was analyzed according to a self-efficacy theory so that to learn about their self-efficacy level regarding problem solving. The researcher believes that motivation and the fact that students chose to study IT, are two of the factors influencing programming self-efficacy. Also in the conclusion part they state problem solving instruction has advantages for programming self-efficacy besides if problem solving self-efficacy level is high as a result individuals insist and put more effort to accomplish success in programming.

Another related research was done by Ozyurt (2015), about distance education and computer programming self-efficacy. Most of the research questions of this study was concerning computer programming self-efficacy. The participants were 104 university students studying computer programming program via distance education. The researcher used computer programming self-efficacy inventory scale which had 9 items with 7-point Likert scale as responses. The Final conclusion of this research is the existence of a positive relationship between student's viewpoint about programming and their programming self-efficacy. Also, the programming self-efficacy of the participants in this study is interpreted as high. Furthermore, there was notable difference in the self-efficacy level of students regarding grade level and gender in other words, concerning gender, males and concerning grade level, second year students had higher computer programming self-efficacy level.

Gökçearslan, Günbatar & Kukul (2017), investigated the development of a Computer Programming Self-Efficacy Scale for secondary school students in order to clarify the self-efficacy levels of youthful students. They generated an item pool which included 30 items with a 5-point Likert scale for the participants to state their conformity level concerning the enquiries in the scale in other words the participants should choose between strongly agree, agree, undecided, disagree and strongly disagree to rate their confidence in doing the computer programming related tasks mentioned in the items. After some specialists looked over the primary item pool, and the exploratory factor analysis the researchers modified the scale to a total of 31 items. As a result of this study an instrument was developed to measure the self-efficacy levels of secondary school students regarding computer programming which Soykan & Kanbul, (2018) used as a data collection tool analyzing K12 Students self-efficacy regarding coding education, and as a result of both studies there was no contrast concerning gender and overall scores acquired from the coding self-efficacy scale. This scale was also used for a research by Yildiz Durak (2018), on impacts of digital story design applications in teaching programming on programming self-efficacy.

Hilal, Suzastri & Zulhazlin (2018), conducted a research on computer programming self-efficacy. The methodology of this research was quantitative, a survey research using a twenty-one item questionnaire to evaluate students' self-efficacy. By approaching a technique called simple random sampling, the participants were 48 nontechnical bachelor students who used Scratch programming-which is a visual programming language. The participants were 42% male and 58% female students. The results showed that firstly the overall self-efficacy of these students was high, secondly, there was no major difference between genders regarding computer programming self-efficacy or computer background and third, there was no relationship between educational achievement of the students and their ability for Scratch programming.

Tsai, Wang & Hsu (2018), investigated the elements influencing computer programming and the development of its scale. They created a computer programming scale including 16 items with 5 subscales as follow: 1)Logical Thinking 2)Cooperation 3)Algorithm 4)Control 5)Debug. This study also showed that between computer programming experience and computer programming self-efficacy, there is an encouraging connection.

In addition, there was a research performed by Soykan & Kanbul, (2018) in North Cyprus related to this subject. The aim of that research was to study and find out k12 students' coding self-efficacy level. The methodology used in this study is descriptive survey and a scale with thirty-one items was used. The main research questions of this study included the relationship between self-efficacy and coding education, gender, age, type of computer being used. The sample of the study was chosen by simple random sampling method which lead to 193 sixth and seventh-grade students. The tools used for gathering data were a personal survey developed by the researcher and the coding self-efficacy scale established by Gökçearslan, Günbatar & Kukul (2017). The results displayed that among students who have not taken a coding course, most of them are willing to take programming lessons, moreover 177 students (91%) would rather using portable computers than desktop, additionally the majority of them devoted beyond two hours of their time to computers. According to the responses for the scale, there is no major difference between the coding self-efficacy level of males and females and the total average score was 2.51 ± 0.55 which means the programming self-efficacy level of these students was at a mediocre level. Also similar to other studies in this field, students who had taken coding education had higher coding self-efficacy in comparison with student who didn't pass programming courses.

.

Chapter 3

METHODOLOGY

In this chapter the research method, the way of gathering all the information needed for this study and data analysis phase are described.

3.1 Research method

The research method of this study is quantitative method. In contrast with a qualitative method, in quantitative method fortunately there is no need for gathering, evaluating and inferring the data simultaneously. Quantitative research is the systematic empirical study of observable occurrences through numerical, mathematical or computational techniques. In this type of research, the researcher wants to check the verification of a theory or relationship (Given, 2008; Neuman, 2014). On the word of Neuman (2014), one of the differences a qualitative research and a quantitative have, is the fact that in the former the researcher uses soft data which includes words, sentences and photos; but in a quantitative research, the scholar uses hard data and numbers. Generally the word data refers to numerical and non-numerical documents that someone collects with respect to some rules (Neuman, 2014). For the time the researcher expects a quantitative data, one of the techniques of gathering information is a survey. In a survey research, same questions are asked from several individuals and their responses get recorded to use the statistics (Neuman, 2014). According to Given, (2008) quantitative methodologies are not suitable to describe the causes of some specified events but they help to rationalize with mathematical perceptions (Neuman, 2014).

Using quantitative method means to compress the gathered data so as to see a greater scope of the event the researcher is studying (Neuman, 2014).

3.2 Participants

This study was about computer programming self-efficacy levels of engineering bachelor students in Eastern Mediterranean University (EMU). EMU has five engineering departments. The total number of engineering bachelor students registered for fall 2018/2019 semester at EMU was 2,398. According to self-selection sampling which is gathering information from people who respond (Saunders, Lewis, & Thornhill, 2009), the participants were 510 engineering bachelor students, studying at EMU. Therefore, the subjects in this study were first, second, third and fourth year engineering bachelor students from the Computer Engineering, Electrical & Electronic Engineering, Mechanical Engineering, Industrial Engineering and Civil Engineering departments who were all registered at EMU, Famagusta, Cyprus.

Table 3.1 shows the distribution of participants with respect to gender, department and class level (grade).

Table 3.1: Demographic information of students

		N	**%**
Gender	Female	105	20.6
	Male	405	79.4
Departments	Computer Engineering	125	24.5
	Electrical and Electronic Engineering	60	11.8
	Mechanical Engineering	81	15.9
	Industrial Engineering	93	18.2
	Civil Engineering	151	29.6
Class Level (grade)	1st Year	69	13.5
	2nd Year	131	25.7
	3rd Year	179	35.1
	4th Year	131	25.7

As it is displayed in the Table 1, 20.6% of the participants were female and 79.4% were male also since the number of female volunteers are less than males this research should not be generalized to both genders. They were studying in fields from 5 different department including Computer Engineering (24.5%), Electrical and Electronic Engineering (11.8%), Mechanical Engineering (15.9%), Industrial Engineering (18.2%) and Civil Engineering (29.6%). While analyzing the data it was understood that students were ranged from freshman (13.5%) to seniors (25.7%) (freshman, sophomore 25.7%, junior 35.1%, senior).

3.3 Data collection tool

According to Marakas et al., (1998) there are only a few methods which can efficiently determine and estimations self-efficacy level. In line with using the survey technique, a written questionnaire was used in this study that is called a computer programming self-efficacy which was developed according to the computer programming self-efficacy scale of Ramalingam & Wiedenbeck (1998); by Askar & Davenport (2009) for their research which was about self-efficacy for Java programming among engineering students. The questionnaire used for this research included 32 items with a 7-point Likert scale, and it contained two sections. The first part was demographic section which consisted questions about gender, department and class level (grade). The second part included the computer programming self-efficacy questions which had 32 items. (see Appendix A).

3.4 Data analysis

In the phase of data analysis researchers usually need to use a software to control the numerical information and convert them to charts and tables (Neuman, 2014). For analyzing the quantitative information for this research, descriptive statistics and Statistical Package for Social Sciences-SPSS was used which showed that Independent sample t-test was applied to calculate the significance of the differences between genders. One-way ANOVA test was used to see if there are major differences between computer programming self-efficacy levels of engineering bachelor students and the variables. The level of quantitative data significance was taken as $p < 0.05$, since as said by Cramer & Howitt (2004), a dissimilarity or connection is probably happening by chance of 5 or fewer times out of 100.

3.5 Reliability and validity

Reliability and validity are methods for indicating the study's thoroughness and the trustworthiness of its results (Roberts, Priest & Traynor, 2006). For using a computer programming self-efficacy scale, the reliability should be calculated by Cronbach's alpha and if reliability quantities are low the items of the scale should be rewritten (Bandura, 2006). For this study the data was collected via Computer Programming Self-Efficacy Scale (CPSES) designed by Ramalingam & Wiedenbeck (1998), for which the calculated Cronbach alpha was around 0.92 in previous studies.

In this study the strength of self-efficacy will be evaluated by answers on a 7-point Likert-type scale ranging from 1 (not confident at all) to 7 (absolutely confident). For questionnaires with several items, is common to use Cronbach's alpha in place of a calculating the internal consistency (Bonett & Wright, 2015). The alpha coefficient reliability for the survey used in this study (the computer programming self-efficacy scale) was 0.98, designating the scale was highly dependable and the items were consistent since it is a number close to 1.

In a quantitative research validity can be described by means of the degree that a theory is calculated correctly (Heale & Twycross, 2015), thus it can be said that this study is valid because it measured computer programming self-efficacy level of the participants.

Chapter 4

FINDINGS AND DISCUSSIONS

This chapter is about the results of this research for which the collected data were descriptively studied with the help of one-way ANOVA, mean analysis and t-test, using SPSS, to find out the level of Computer Programming Self-Efficacy (CPSE) of the students and see if there is a significant difference between the genders or among different departments or class level of the participants.

Table 4.1: Descriptive information of the scale according to the responses

Mean	129.55
Std. Deviation	51.58
Minimum	32
Maximum	224

As it's shown on Table 4.1, the maximum score a person could get was 224 for students who were absolutely confident in all thirty-two tasks mentioned in items and the minimum was 32, since there were thirty-two items with seven options as a 7-point Likert scale. The total score each participant had was calculated and the average score was 129.55, besides there were 7 students out of 510; who scored the maximum 224, and 10 students scored the minimum 32. Out of those seven people who scored the maximum score, one was studying mechanical engineering, one was from electrical and electronics engineering department and five were from computer engineering department.

For each item the total score from all students could be 3570 but none of the items got a 7-point mark from all participants. The maximum total score (2522) was for item 5 which indicates the volunteers were more confidence in writing a program that computes the average of three numbers than other tasks in the items. Furthermore, the minimum score (1788) was for item number 11 that shows there was a low level of programming self-efficacy aimed at writing a long and complex program to solve any given problem as long as the specifications are clearly defined.

4.1 CPSE level of engineering students

In this section the computer programming self-efficacy level of engineering undergraduate students at EMU was evaluated. Table 4.2 shows in number and percentages, how the students rated their confidence in other words their level of self-efficacy for each item according to the scale of 1 to 7.

Table 4.2: Engineering students' self-efficacy level

	1-Not confident at all		2-Mostly not confident		3-Slightly confident		4- Neutral		5-Fairly confident		6-Mostly confident		7-Absolutely confident		Mean	Std. Deviation
	N	%	N	%	N	%	N	%	N	%	N	%	N	%		
I1	77	15%	58	11%	68	13%	**94**	**18%**	93	18%	54	11%	66	13%	3.96	1.92
I2	61	12%	44	9%	70	14%	95	19%	**96**	**19%**	84	16%	60	12%	4.20	1.85
I3	73	14%	54	11%	68	13%	84	16%	**90**	**18%**	68	13%	73	14%	4.09	1.95
I4	60	12%	39	8%	46	9%	69	14%	70	14%	52	10%	**174**	**34%**	**4.76**	2.11
I5	57	11%	35	7%	36	7%	58	11%	73	14%	67	13%	**184**	**36%**	**4.94**	2.09
I6	51	10%	38	7%	40	8%	77	15%	74	15%	65	13%	**165**	**32%**	**4.84**	2.03
I7	78	15%	49	10%	62	12%	78	15%	87	17%	67	13%	**89**	**17%**	4.18	2.03
I8	**150**	**29%**	51	10%	54	11%	73	14%	61	12%	56	11%	65	13%	**3.53**	2.15
I9	78	15%	53	10%	72	14%	70	14%	72	14%	61	12%	**104**	**20%**	4.18	2.08
I10	**109**	**21%**	74	15%	52	10%	73	14%	87	17%	72	14%	43	8%	3.67	2.00
I11	**140**	**27%**	57	11%	64	13%	59	12%	76	15%	72	14%	42	8%	**3.50**	2.06
I12	**125**	**25%**	66	13%	59	12%	60	12%	78	15%	72	14%	50	10%	**3.61**	2.07
I13	**128**	**25%**	63	12%	49	10%	73	14%	82	16%	61	12%	54	11%	3.62	2.07
I14	**106**	**21%**	64	13%	58	11%	73	14%	73	14%	74	15%	62	12%	3.80	2.06
I15	**97**	**19%**	61	12%	50	10%	66	13%	80	16%	82	16%	74	15%	4.00	2.09
I16	**115**	**23%**	53	10%	67	13%	64	13%	73	14%	72	14%	66	13%	3.79	2.10

I17	**98**	**19%**	61	12%	44	9%	75	15%	79	15%	77	15%	76	15%	4.00	2.09
I18	**130**	**25%**	64	13%	53	10%	72	14%	77	15%	68	13%	46	9%	**3.56**	2.05
I19	67	13%	43	8%	48	9%	68	13%	**98**	**19%**	91	18%	95	19%	**4.45**	2.00
I20	70	14%	52	10%	68	13%	81	16%	**85**	**17%**	80	16%	74	15%	4.16	1.96
I21	73	14%	49	10%	45	9%	65	13%	85	17%	93	18%	**100**	**20%**	4.40	2.06
I22	73	14%	42	8%	51	10%	69	14%	97	19%	79	15%	**99**	**19%**	4.38	2.03
I23	77	15%	48	9%	45	9%	52	10%	94	18%	85	17%	**109**	**21%**	4.42	2.10
I24	74	15%	63	12%	54	11%	81	16%	84	16%	**87**	**17%**	67	13%	4.11	1.98
I25	81	16%	51	10%	55	11%	68	13%	**92**	**18%**	91	18%	72	14%	4.17	2.01
I26	**98**	**19%**	62	12%	61	12%	71	14%	92	18%	79	15%	47	9%	3.82	1.98
I27	**89**	**17%**	54	11%	59	12%	77	15%	82	16%	79	15%	70	14%	4.03	2.03
I28	**118**	**23%**	63	12%	53	10%	72	14%	81	16%	76	15%	47	9%	3.68	2.04
I29	**124**	**24%**	47	9%	58	11%	76	15%	77	15%	77	15%	51	10%	3.72	2.06
I30	**94**	**18%**	56	11%	63	12%	65	13%	93	18%	77	15%	62	12%	3.95	2.02
I31	**90**	**18%**	61	12%	59	12%	74	15%	86	17%	75	15%	65	13%	3.96	2.02
I32	**108**	**21%**	60	12%	38	7%	76	15%	89	17%	74	15%	65	13%	3.90	2.08

According to Table 4.2 if the standard deviation is small, it means that a lot of values are close to the mean. The minimum standard deviation in this study is for item 2 which indicates most participants (19%) are fairly confident about their ability to understand the language structure of computer programming languages and the usage of the reserved words. The peak of standard deviation is 2.15 for item 8 which is about participants believing they have the ability to build their own application, and the mean is 3.53, that outlines the responses had considerable difference since 29% marked they are not confident at all and 13% thought they are absolutely confident.

The four maximum means were 4.94, 4.84, 4.76, 4.45 for items 4,5,6 and 19 which by paying attention to the responses, it can be implied that most of the students had high self-efficacy level in these tasks**.** Among them the highest mean (4.94) was for item 5 and it showed the participants (36%) had more confidence in writing a program that computes the average of three numbers than other tasks mentioned on other items of the survey. On the other hand, four smallest numbers among means were 3.50, 3.53, 3.56, 3.61, which comparing to the responses, it showed for items 8,11,18 and 12 the majority of student's computer programming self-efficacy was low. Moreover, the lowest mean (3.50) was for item 11, which implies that comparing to other tasks, the students didn't believe in being able to write a long and complex program to solve any given problem as long as the specifications were clearly defined; but the standard deviation for the fifth item was 2.09 and for item 11 was 2.06 which were not small numbers.

Overall, the maximum mean was 4.94 and the minimum mean was 3.50, besides the average of all items means was 4.04 so the computer programming self-efficacy of these students cannot be interpreted as high since in the responses scale number 4 was for feeling 50/50 confident (neutral) and number 3 for being slightly confident.

Thus it can be said that the CPSE level of these students was at an average level which is like the result Soykan & Kanbul (2018) got; however, the programming self-efficacy of the participants in Ozyurt (2015) and Hilal et al. (2018) study was taken to be high.

4.2 CPSE level of engineering students according to gender

To investigate the relation between computer programming self-efficacy and gender an independent sample t-test was used.

Table 4.3: Descriptive statistics for CPSE level regarding the gender

	Gender	N	Mean	Std. Deviation	Sig.	T	df	Sig. (2-tailed)
I1	Female	105	4.02	1.87	**0.26**	0.35	508	0.72
	Male	405	3.95	1.94				
I2	Female	105	4.38	1.64	0.01	1.20	182.40	0.22
	Male	405	4.15	1.9				

I3	Female	105	4.4	1.75	0.04	1.98	181.13	0.04
	Male	405	4.01	2.00				
I4	Female	105	4.85	1.83	0.01	0.53	188.58	0.63
	Male	405	4.74	2.18				
I5	Female	105	4.93	1.85	0.01	-0.07	183.37	0.94
	Male	405	4.94	2.15				
I6	Female	105	**5**	1.64	0.01	1.02	203.30	0.3
	Male	405	4.8	2.12				
I7	Female	105	4.47	1.88	**0.08**	1.65	508	0.09
	Male	405	4.1	2.06				
I8	Female	105	4.04	1.9	0.01	3.00	182.86	0.01
	Male	405	3.4	2.19				
I9	Female	105	4.34	1.94	**0.06**	0.87	508	0.38
	Male	405	4.14	2.12				
I10	Female	105	4.2	1.85	0.01	3.27	173.30	0.01
	Male	405	3.53	2.01				
I11	Female	105	4.17	1.87	0.01	3.99	176.39	0
	Male	405	**3.33**	2.07				
I12	Female	105	4.27	1.90	0.01	3.88	174.38	0
	Male	405	3.44	2.08				
I13	Female	105	4.27	1.88	0.01	3.89	175.93	0
	Male	405	3.45	2.08				
I14	Female	105	4.34	1.88	0.01	3.18	176.92	0.01
	Male	405	3.67	2.09				
I15	Female	105	4.38	1.89	0.01	2.21	179.22	0.02
	Male	405	3.9	2.13				
I16	Female	105	4.36	1.88	0.01	3.34	179.69	0.01
	Male	405	3.65	2.13				
I17	Female	105	4.41	1.94	0.03	2.41	174.01	0.01
	Male	405	3.89	2.12				
I18	Female	105	4.34	1.83	0.01	4.71	177.92	0
	Male	405	3.36	2.06				
I19	Female	105	4.71	1.81	0.01	1.62	178.83	0.1
	Male	405	4.38	2.04				
I20	Female	105	4.42	1.8	**0.07**	1.53	508	0.12
	Male	405	4.09	2.00				
I21	Female	105	4.78	1.81	0.01	2.27	184.48	0.02
	Male	405	4.31	2.11				
I22	Female	105	4.64	1.85	0.02	1.57	177.97	0.11
	Male	405	4.32	2.07				
I23	Female	105	4.74	1.79	0.01	1.92	191.05	0.05
	Male	405	4.34	2.17				

I24	Female	105	4.69	1.86	**0.18**	3.42	508	0.01
	Male	405	3.96	1.98				
I25	Female	105	4.76	1.77	0.01	3.66	182.58	0.01
	Male	405	4.02	2.05				
I26	Female	105	4.49	1.87	**0.10**	3.92	508	0
	Male	405	3.65	1.97				
I27	Female	105	4.68	1.75	0.01	4.13	186.62	0
	Male	405	3.86	2.06				
I28	Female	105	4.52	1.97	**0.18**	4.80	508	0
	Male	405	3.47	2.00				
I29	Female	105	4.47	1.93	**0.09**	4.25	508	0
	Male	405	3.53	2.05				
I30	Female	105	4.6	1.82	0.01	4.03	177.62	0
	Male	405	3.78	2.04				
I31	Female	105	4.6	1.76	0.01	4.02	184.29	0
	Male	405	3.79	2.05				
I32	Female	105	4.48	1.84	0.01	3.53	182.41	0.01
	Male	405	3.75	2.12				

Using a t-test for the relation between the gender and the level of self-efficacy showed that for 8 items out of 32 the significance level is more than 0.05. According to the significance and upper-lower point of confidence intervals of difference, females had a higher self-efficacy in 8 items. In essence, it can be said that for tasks like writing syntactically correct coding statements (Item 1), using built-in functions in programming software (Item 7), writing a minor program according to a small familiar problem (Item 9), finalizing a programming project while having the language reference manual (Item 20), males have a lower computer programming self-efficacy. Similarly, females have higher CPSE level while doing tasks like completing a programming project if the built-in help facility was available (Item 24), shortly having a suitable strategy for a programming project (Item 26), conceptually tracing the implementation of a long, multi-file program (Item 28), rewriting confusing parts of code in order to make it more legible and understandable (Item 29). However, since the number of female participants were just 105, these findings about the gender should not be generalized but for 25% of the items females had a higher CPSE level.

This is contradictory to the result of Askar & Davenport (2009), which indicated self-efficacy of males was higher than that of females for Java programming, also Ozyurt (2015), found out that males CPSE was higher than females; however as a result of some other researches there was no significant difference between the genders regarding coding self-efficacy (Gökçearslan et al., 2017; Hilal et al., 2018; Soykan & Kanbul, 2018).

4.3 CPSE level of Students regarding engineering departments

As it can be seen in tables bellow, to realize if there is a correlation between the Computer Programming Self-efficacy level of Engineering students and their Department a One Way ANOVA test and Post Hoc comparison were computed. In following tables K stands for Computer Engineering, E for Electrical and Electronic Engineering, M for Mechanical Engineering, I for Industrial Engineering and C for Civil Engineering.

Table 4.4: Descriptive statistics for writing syntactically correct coding statements regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.24	1.54
Electrical and Electronic Engineering	60	4.68	1.78
Mechanical Engineering	81	3.66	1.79
Industrial Engineering	93	3.86	1.64
Civil Engineering	151	2.86	1.78

Table 4.5: ANOVA summery for writing syntactically correct coding statements depending on department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	426.45	4	106.61	36.64	0.01	K-E, K-M, K-I, K-C
Within Groups	1469.04	505	2.90			E-M, E-I, E-C
Total	1895.49	509				M-C I-C

The significance in ANOVA (Table 4.5) for item 1 was less than 0.05, and the mean values are different which points out a meaningful difference among Departments regarding computer programming self-efficacy and [F(4,505)= 36.64, p=0.000]. On the other hand, Post Hoc comparison reveals that for item 1 there is a relation between departments and level of computer programming self-efficacy. Since the mean value of computer engineering students is 5.24 with a standard deviation of 1.54 there was a major difference among students studying computer engineering with all other departments including Electrical and Electronic Engineering (Mean=4.68, S.D.=1.78), Mechanical Engineering (Mean=3.66, S.D.=1.79), Industrial Engineering (Mean=3.86, S.D.=1.64), and Civil Engineering (Mean=2.86, S.D.=1.78). Furthermore, the mean score of item 1 for Electrical and Electronic Engineering students (Mean=4.68, S.D.=1.78), differed meaningfully from the Mechanical Engineering (Mean=3.66, S.D.=1.79), Industrial Engineering (Mean=3.86, S.D.=1.64) and Civil Engineering (Mean=2.86, S.D.=1.78) students. In addition, there was also a difference between the mean results of students who study Mechanical Engineering with the one’s studying Civil Engineering. Moreover, Industrial Engineering mean score did not differ significantly from the one for Mechanical Engineering; however, it did vary from the mean of Civil Engineering.

Table 4.6: Descriptive statistics for understanding the programming languages structure regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.29	1.48
Electrical and Electronic Engineering	60	4.91	1.64
Mechanical Engineering	81	4.06	1.61
Industrial Engineering	93	4.19	1.68
Civil Engineering	151	3.09	1.78

Table 4.7: ANOVA summery for understanding the programming languages structure regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	367.65	4	91.91	33.67	0.01	K-M, K-I, K-C
Within Groups	1378.54	505	2.73			E-M, E-I, E-C
Total	1746.19	509				M-C I-C

One way ANOVA was performed to expose any relationship between the departments and computer programming self-efficacy level of students. As for item 2, it can be seen from Table 4.6 that there is an important dissimilarity of department on Computer Programming Self-Efficacy level of students(sig<0.05), [F(4,505)= 33.671, p=0.000]. For item 2, between students at computer engineering (Mean=5.29, S.D.= 1.48) and other departments mean value, there is a huge dissimilarity with Mechanical Engineering (Mean=4.06, S.D.=1.615) and Industrial Engineering (Mean=4.19, S.D.=1.68) and Civil Engineering (Mean=3.09, S.D.= 1.78). But there is a similarity between the mean of computer engineering students and Electrical and Electronic Engineering (Mean=4.91, S.D.=1.64) students. What's more is that, the mean score of item 2 for Electrical and Electronic Engineering students (Mean=4.91, S.D.=1.64), differed meaningfully from the Mechanical Engineering (Mean=4.06, S.D.=1.61), Industrial Engineering (Mean=4.19, S.D.=1.68) and Civil Engineering (Mean=3.09, S.D.=1.78) students. Likewise, the mean score for Mechanical Engineering has a different amount than the one for Civil Engineering. On the other hand, mean score of Industrial Engineering is similar to Mechanical Engineering but it has a significant difference with Civil Engineering.

Table 4.8: Descriptive statistics for writing logically correct blocks of code regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.28	1.57
Electrical and Electronic Engineering	60	5.11	1.68
Mechanical Engineering	81	4.03	1.81
Industrial Engineering	93	4.12	1.65
Civil Engineering	151	2.72	1.71

Table 4.9: ANOVA summery for writing logically correct blocks of code regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	520.507	4	130.12	45.807	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1434.591	505	2.841			
Total	1955.098	509				

According to Table 4.9, [F(4,505)= 45.80, p=0.01] and the mean value of item 3 for Computer Engineering (Mean=5.28, S.D.=1.57) and Electrical and Electronic Engineering (Mean=5.11, S.D.=1.68) are close to each other. However, the Computer Engineering mean differs significantly with other three departments including Mechanical Engineering (Mean=4.03, S.D.=1.81), Industrial Engineering (Mean=4.12, S.D.=1.65) and Civil Engineering (Mean=2.72, S.D.=1.71). The score mean of Electrical and Electronic Engineering students for item 3, has a major difference with three departments of Mechanical Engineering, Industrial Engineering and Civil Engineering. The data given by Mechanical Engineering students showed a mean value which is meaningfully different with the one from Civil Engineering department. Although the mean score of Industrial Engineering (Mean=4.12, S.D.= 1.65) is similar to the one from Mechanical Engineering but it has a major difference with Civil Engineering (Mean=2.72, S.D.=1.71).

Table 4.10: Descriptive statistics for writing a program that displays a greeting message regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	6.08	1.46
Electrical and Electronic Engineering	60	5.18	1.83
Mechanical Engineering	81	4.65	2.01
Industrial Engineering	93	4.69	1.86
Civil Engineering	151	3.62	2.22

Table 4.11: ANOVA summery for writing a program that displays a greeting message regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	425.14	4	106.28	28.86	0.01	K-E, K-M, K-I, K-C
Within Groups	1859.55	505	3.68			E-C
Total	2284.69	509				M-C I-C

For item 4 (see Table 4.11) [F(4,505)= 28.86, p=0.01] and the mean score for Computer Engineering students (Mean=6.08, S.D.=1.46) has an essential difference with all four departments which are Electrical and Electronic Engineering (Mean=5.18, S.D.=1.83) Mechanical Engineering (Mean=4.65, S.D.=2.01), Industrial

Engineering (Mean=4.69, S.D.=1.86) and Civil Engineering (Mean=3.62, S.D.=2.22). In addition, mean score for Electrical and Electronic Engineering similar to the mean of Mechanical Engineering students and Industrial Engineering students, however it differs expressively with the mean of Civil Engineering. Moreover, the mean value of Mechanical Engineering students has a major difference comparing to the one for Civil Engineering. Besides for item 4, Industrial Engineering mean score is very close to the mean of Mechanical Engineering students but it there exists an important difference between the mean of Industrial Engineering and Civil Engineering.

Table 4.12: Descriptive statistics for writing a program that computes the average of three numbers regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	6.14	1.42
Electrical and Electronic Engineering	60	5.53	1.85
Mechanical Engineering	81	4.70	2.03
Industrial Engineering	93	5.10	1.88
Civil Engineering	151	3.74	2.14

Table 4.13: ANOVA summery for writing a program that computes the average of three numbers regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	423.87	4	105.96	29.62	0.01	K-E, K-M, K-I, K-C
Within Groups	1806.59	505	3.57			E-M, E-C
Total	2230.46	509				M-C , I-C

As for item 5 by analyzing Table 4.12, Table 4.13 and Post Hoc comparison, it can be said that [F(4,505)= 29.62, p=0.01] and among the mean of Computer Engineering students (Mean=6.14, S.D.=1.42) and other departments there is an important difference because the mean value for Electrical and Electronic Engineering is 5.53

(S.D.=1.85), for Mechanical Engineering is 4.70 (S.D.=2.03), for Industrial Engineering is 5.10 (S.D.=1.88) and for Civil Engineering is 3.74 (S.D.=2.14). Furthermore, the mean score of Electrical and Electronic Engineering differs significantly with both means of Mechanical and Civil Engineering however it is a number close to the mean of Industrial Engineering. On the other hand, the mean value of Mechanical Engineering differs meaningfully with the one for Civil Engineering. Additionally, the mean value of Industrial Engineering is similar to the mean for Mechanical Engineering students, yet between Industrial Engineering mean and Civil Engineering there is a significant difference.

Table 4.14: Descriptive statistics for coding a program to find out the average some numbers regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	6.10	1.34
Electrical and Electronic Engineering	60	5.41	1.72
Mechanical Engineering	81	4.58	1.88
Industrial Engineering	93	5.10	1.72
Civil Engineering	151	3.54	2.10

Table 4.15: ANOVA summery for coding a program to find out the average some numbers regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	483.18	4	120.79	37.55	0.01	K-E, K-M, K-I, K-C E-M, E-C M-C I-C
Within Groups	1624.26	505	3.21			
Total	2107.45	509				

Table 4.15 displays that [F(4,505)=37.557, p=0.01]. Additionally, for item 6, the mean value for Computer Engineering students (Mean=6.10, S.D.=1.34) has an important difference with all four departments which are Electrical and Electronic Engineering (Mean=5.41, S.D.=1.72) Mechanical Engineering (Mean=4.58, S.D.=1.88), Industrial Engineering (Mean=5.10, S.D.=1.72) and Civil Engineering (Mean=3.54, S.D.=2.106). Moreover, the mean score of Electrical and Electronic Engineering students for item 6 is similar to the one for Industrial Engineering, however there is a substantial difference between the mean of Electrical and Electronic Engineering and two other departments which are Mechanical Engineering and Civil Engineering. What is more is that the mean value of Mechanical Engineering department differs notably with the one from Civil Engineering (Mean=3.54, S.D.=2.106). On one hand, Industrial Engineering mean value is a close number to the mean from Mechanical Engineering, on the other there exists a major difference between the mean score of Industrial Engineering and Civil Engineering.

Table 4.16: Descriptive statistics for using built-in functions in a programming software regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.50	1.64
Electrical and Electronic Engineering	60	5.20	1.64
Mechanical Engineering	81	4.01	1.90
Industrial Engineering	93	3.96	1.70
Civil Engineering	151	2.91	1.85

Table 4.17: ANOVA summery for using built-in functions in a programming software regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	530.05	4	132.51	42.66	0.01	K-M, K-I, K-C

Within Groups	1568.62	505	3.106	E-M, E-I, E-C M-C I-C
Total	2098.67	509		

For item 7, by paying attention to Table 4.16 and Table 4.17, it can be said that [F(4,505)=37.55, p=0.01]. Also the mean score of Computer Engineering students (Mean=5.50, S.D.=1.64) is a very close number to the one from Electrical and Electronic Engineering (Mean=5.20, S.D.=1.64), besides it has an important difference with other three departments which are Mechanical Engineering (Mean=4.01, S.D.=1.90), Industrial Engineering (Mean=3.96, S.D.=1.70) and Civil Engineering (Mean=2.91, S.D.=1.85). Moreover, Electrical and Electronic Engineering mean differs considerably from the mean of three departments including Mechanical, Industrial and Civil Engineering. Additionally, the mean score of Mechanical Engineering (Mean=4.01, S.D.=1.90) is similar to the one from Industrial Engineering department but it has an essential difference with Civil Engineering (Mean=2.91, S.D.=1.85). On the other hand, the mean score of Industrial Engineering students extensively differs from Civil Engineering department.

Table 4.18: Descriptive statistics for building their own application regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.68	1.97
Electrical and Electronic Engineering	60	4.45	1.87
Mechanical Engineering	81	3.61	2.10
Industrial Engineering	93	3.39	2.04
Civil Engineering	151	2.25	1.77

Table 4.19: ANOVA summery for building their own application regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	462.541	4	115.63	30.66	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1904.393	505	3.77			
Total	2366.933	509				

Considering table 4.18 and table 4.19 the F value for item 8 is [F(4,505)= 30.66, p=0.01]. In addition, the mean score of Computer Engineering students (Mean=4.68, S.D.=1.97) is a very close number to the one from Electrical and Electronic Engineering (Mean=4.45, S.D.=1.87), besides it has a noticeable difference with other three departments which are Mechanical Engineering (Mean=3.61, S.D.=2.10), Industrial Engineering (Mean=3.39, S.D.=2.04) and Civil Engineering (Mean=2.25, S.D.=1.77). Besides, Electrical and Electronic Engineering mean differs considerably from the mean of three departments including Mechanical, Industrial and Civil Engineering. Furthermore, the mean score of Mechanical Engineering (Mean=3.61, S.D.=2.10), is similar to the one from Industrial Engineering department but it has an essential difference with the mean from Civil Engineering department. Then again, the mean score of Industrial Engineering students (Mean=3.39, S.D.=2.04) extensively differs from Civil Engineering department.

Table 4.20: Descriptive statistics for writing a small program according to a small familiar problem regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.40	1.82
Electrical and Electronic Engineering	60	5.08	1.59
Mechanical Engineering	81	4.08	1.92
Industrial Engineering	93	4.06	1.85
Civil Engineering	151	2.94	1.95

Table 4.21: ANOVA summery for writing a small program according to a small familiar problem regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	471.42	4	117.85	34.02	0.01	K-M, K-I, K-C
Within Groups	1749.24	505	3.46			E-M, E-I, E-C
Total	2220.67	509				M-C I-C

In view of Table 4.21 the F value for item 9 is [F(4,505)= 34.02, p=0.01]. Furthermore, the mean score of Computer Engineering students (Mean=5.40, S.D.= 1.82) is an actual close number to the one from Electrical and Electronic Engineering (Mean=5.08, S.D.=1.59), also it has a visible difference with other three departments which are Mechanical Engineering (Mean=4.08, S.D.=1.92), Industrial Engineering (Mean=4.06, S.D.=1.85) and Civil Engineering (Mean=2.94, S.D.=1.95). In addition, Electrical and Electronic Engineering mean differs extensively from the mean of three departments including Mechanical, Industrial and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department (Mean=4.06, S.D.=1.85) but it has a notable difference with the mean from Civil Engineering department (Mean=2.94, S.D.=1.95). Nevertheless, the mean score of Industrial Engineering students suggestively differs from Civil Engineering department.

Table 4.22: Descriptive statistics for writing a program to solve a problem regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.70	1.84
Electrical and Electronic Engineering	60	4.83	1.52
Mechanical Engineering	81	3.74	1.95
Industrial Engineering	93	3.55	1.74
Civil Engineering	151	2.39	1.68

Table 4.23: ANOVA summery for writing a program to solve a problem regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	463.50	4	115.87	37.20	0.01	K-M, K-I, K-C
Within Groups	1572.80	505	3.11			E-M, E-I, E-C
Total	2036.31	509				M-C I-C

Given the information in Table 4.22 and Table 4.23 the F value for item 10 is [F(4,505)= 37.206, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.70, S.D.= 1.84) has a detectable difference with three departments including Mechanical Engineering (Mean=3.74, S.D.=1.95), Industrial Engineering (Mean=3.55, S.D.=1.74) and Civil Engineering (Mean=2.39, S.D.=1.68). Besides, the mean score of Electrical and Electronic students (Mean=4.83, S.D.=1.52) is a similar to the one from Computer Engineering (Mean=4.70, S.D.=1.84), on the contrary it has an observable difference with other three departments which are Mechanical Engineering (Mean=3.74, S.D.=1.95), Industrial Engineering (Mean=3.55, S.D.= 1.74) and Civil Engineering (Mean=2.39, S.D.=1.68). Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean (Mean=3.55, S.D.=1.74) has difference comparing to Civil Engineering (Mean=2.39, S.D.=1.68).

Table 4.24: Descriptive statistics for writing a program while the specifications are defined regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.56	1.91
Electrical and Electronic Engineering	60	4.73	1.77

Mechanical Engineering	81	3.59	1.95
Industrial Engineering	93	3.37	1.97
Civil Engineering	151	2.17	1.55

Table 4.25: ANOVA summery for writing a program while the specifications are defined regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	502.17	4	125.54	38.02	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1667.31	505	3.30			
Total	2169.48	509				

As for item 11, Table 4.24 and Table 4.25 show [F(4,505)= 38.025, p=0.01]. What's more is that the mean score of Computer Engineering students (Mean=4.56, S.D.=1.91) has an obvious difference with three departments including Mechanical Engineering (Mean=3.59, S.D.=1.95), Industrial Engineering (Mean=3.37, S.D.=1.97) and Civil Engineering (Mean=2.17, S.D.=1.55). Moreover, the mean score of Electrical and Electronic students (Mean=4.73, S.D.=1.77) is similar to the one from Computer Engineering, yet differs from the mean of other three departments which are Mechanical, Industrial and Civil Engineering departments. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs extensively from the mean of Civil Engineering department. Then again, Industrial Engineering mean has difference comparing to the Civil Engineering mean.

Table 4.26: Descriptive statistics for designing a program in a modular manner regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.71	1.89
Electrical and Electronic Engineering	60	4.65	1.74
Mechanical Engineering	81	3.64	1.97
Industrial Engineering	93	3.75	1.94
Civil Engineering	151	2.21	1.62

Table 4.27: ANOVA summery for designing a program in a modular manner regarding the department of engineering students

Variable Source	Sum of Squares	Df	Mean Square	F	Sig.	Significant Difference
Between Groups	513.77	4	128.44	38.69	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1676.43	505	3.32			
Total	2190.20	509				

In view of table 4.26 and table 4.27 the F value for item 12 is [F(4,505)= 38.69, p=0.01]. In addition, the mean score of Computer Engineering students (Mean=4.71, S.D.= 1.89) is very close to the one from Electrical and Electronic Engineering (Mean=4.65, S.D.= 1.74), also it has a noticeable difference with other three departments which are Mechanical Engineering (Mean=3.64, S.D.=1.97), Industrial Engineering (Mean=3.75, S.D.=1.94) and Civil Engineering (Mean=2.21, S.D.=1.62). Furthermore, Electrical and Electronic Engineering mean differs extensively from the mean of three departments including Mechanical, Industrial and Civil Engineering. Likewise, the mean score of Mechanical Engineering (Mean=3.64, S.D.=1.97), suggestively differs from Civil Engineering department. Nevertheless, the mean score of Industrial Engineering students (Mean=3.75, S.D.=1.94), is similar to the one from

Mechanical Engineering department but it has a notable difference with the mean from Civil Engineering department (Mean=2.21, S.D.=1.62).

Table 4.28: Descriptive statistics for understanding the object-oriented paradigm regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.62	1.96
Electrical and Electronic Engineering	60	4.73	1.76
Mechanical Engineering	81	3.54	2.01
Industrial Engineering	93	3.51	1.84
Civil Engineering	151	2.45	1.80

Table 4.29: ANOVA summery for understanding the object-oriented paradigm regarding the Department of Engineering students

Variable Source	Sum of Squares	Df	Mean Square	F	Sig.	Significant Difference
Between Groups	406.10	4	101.52	28.70	0.01	K-M, K-I, K-C
Within Groups	1785.85	505	3.53			E-M, E-I, E-C M-C
Total	2191.96	509				I-C

According to Table 4.28 and Table 4.29 the F value for item 13 is [F(4,505)= 28.70, p=0.01]. Furthermore, the mean score of Computer Engineering students (Mean=4.62, S.D.=1.96), has a meaningfully difference with three departments including Mechanical Engineering (Mean=3.54, S.D.=2.01), Industrial Engineering (Mean=3.51, S.D.= 1.84) and Civil Engineering (Mean=2.45, S.D.=1.80). Besides, the mean score of Electrical and Electronic students (Mean=4.73, S.D.=1.76) is very close to the one from Computer Engineering, in contrast it has a recognizable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical

Engineering (Mean=3.54, S.D.=2.01), is similar to the one from Industrial Engineering department (Mean=3.51, S.D.=1.84) but it differs suggestively from the mean of Civil Engineering department (Mean=2.45, S.D.=1.80). Also, Industrial Engineering mean has a notably difference comparing to Civil Engineering.

Table 4.30: Descriptive statistics for identifying and defining the objects in the problem domain regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.78	1.83
Electrical and Electronic Engineering	60	4.85	1.68
Mechanical Engineering	81	3.96	1.98
Industrial Engineering	93	3.81	1.97
Civil Engineering	151	2.50	1.77

Table 4.31: ANOVA summery for identifying and defining the objects in the problem domain regarding the department of engineering students

Variable Source	Sum of Squares	Df	Mean Square	F	Sig.	Significant Difference
Between Groups	443.20	4	110.80	32.28	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1733.34	505	3.43			
Total	2176.55	509				

As stated in Table 4.30 and Table 4.31 the F value for item 14 is [F(4,505)= 32.28, p=0.01], and the mean value of Computer Engineering students (Mean=4.78, S.D.=1.83), has a detectable difference with three departments including Mechanical Engineering, Industrial Engineering (Mean=3.81, S.D.=1.97) and Civil Engineering (Mean=2.50, S.D.=1.77). Besides, the mean score of Electrical and Electronic students (Mean=4.85, S.D.=1.68) is a similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are

Mechanical Engineering (Mean=3.96, S.D.=1.98), Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has difference comparing to Civil Engineering.

Table 4.32: Descriptive statistics for using of a pre-written function while having a labeled declaration of it regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.08	1.91
Electrical and Electronic Engineering	60	5.15	1.64
Mechanical Engineering	81	4.22	1.96
Industrial Engineering	93	3.79	1.83
Civil Engineering	151	2.67	1.84
Total	510	4.005	2.09

Table 4.33: ANOVA summery for using of a pre-written function while having a labeled declaration of it regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	497.91	4	124.47	36.06	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1743.06	505	3.45			
Total	2240.98	509				

Given the information in Table 4.32 and Table 4.33 the F value for item 15 is [F(4,505)=36.06, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.08, S.D.=1.911), has a detectable difference with three departments including Mechanical Engineering (Mean=4.22, S.D.=1.96), Industrial Engineering (Mean=3.79, S.D.=1.83) and Civil Engineering (Mean=2.67, S.D.=1.84). Besides, the mean score of Electrical and Electronic students (Mean=5.15, S.D.=1.64) is a similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has difference comparing to Civil Engineering**.**

Table 4.34: Descriptive statistics for using a defined class while having a labeled declaration of it regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.79	2.005
Electrical and Electronic Engineering	60	4.85	1.72
Mechanical Engineering	81	3.92	1.86
Industrial Engineering	93	3.70	1.94
Civil Engineering	151	2.54	1.86
Total	510	3.79	2.10

Table 4.35: ANOVA summery for using a defined class while having a labeled declaration of it regarding the department of engineering students

Variable Source	Sum of Squares	Df	Mean Square	F	Sig.	Significant Difference
Between Groups	429.76	4	107.44	29.83	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1818.42	505	3.60			
Total	2248.19	509				

According to Table 4.34 and Table 4.35 the F value for item 16 is [F(4,505)=29.83, p=0.01], and the mean score of Computer Engineering students (Mean=4.79, S.D.=2.005), has a detectable difference with three departments including Mechanical Engineering (Mean=3.92, S.D.=1.86), Industrial Engineering (Mean=3.70, S.D.=1.94) and Civil Engineering (Mean=2.54, S.D.=1.86). Besides, the mean score of Electrical and Electronic students (Mean=4.85, S.D.=1.72) is a similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has difference comparing to Civil Engineering.

Table 4.36: Descriptive statistics for debugging a written program regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.00	1.85
Electrical and Electronic Engineering	60	5.16	1.48
Mechanical Engineering	81	4.03	1.97
Industrial Engineering	93	3.80	1.97
Civil Engineering	151	2.81	1.97
Total	510	4.002	2.09

Table 4.37: ANOVA summery for debugging a written program regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	422.45	4	105.61	29.38	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1818.54	505	3.60			
Total	224.99	509				

According to Table 4.36 and Table 4.37 the F value for item 17 is [F(4,505)=29.32, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.00, S.D.=1.85), has a detectable difference with three departments including Mechanical Engineering (Mean=4.03, S.D.=1.97), Industrial Engineering (Mean=3.80, S.D.= 1.97) and Civil Engineering (Mean=2.81, S.D.=1.97). Besides, the mean score of Electrical and Electronic students (Mean=5.16, S.D.=1.48) is a similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has difference comparing to Civil Engineering

Table 4.38: Descriptive statistics for comprehending a multi-file program regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.27	1.85
Electrical and Electronic Engineering	60	4.76	1.72
Mechanical Engineering	81	3.65	2.11
Industrial Engineering	93	3.54	1.83
Civil Engineering	151	2.47	1.91
Total	510	3.56	2.05

Table 4.39: ANOVA summery for comprehending a multi-file program regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	328.59	4	82.14	22.81	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1818.50	505	3.60			
Total	2147.09	509				

According to Table 4.38 and Table 4.39 the F value for item 18 is [F(4,505)=22.81, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.27, S.D.= 1.85), has a detectable difference with three departments including Mechanical Engineering (Mean=3.65, S.D.=2.11), Industrial Engineering (Mean=3.54, S.D.= 1.83) and Civil Engineering (Mean=2.47, S.D.=1.91). Besides, the mean score of Electrical and Electronic students (Mean=4.76, S.D.=1.72) is a similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has difference comparing to Civil Engineering

Table 4.40: Descriptive statistics for completing a program while knowing how to solve the problem regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.22	1.76
Electrical and Electronic Engineering	60	5.05	1.51
Mechanical Engineering	81	4.49	1.82
Industrial Engineering	93	4.61	1.88
Civil Engineering	151	3.45	2.11
Total	510	4.45	2.00

Table 4.41: ANOVA summery for completing a program while knowing how to solve the problem regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	250.008	4	62.50	17.67	0.01	K-M, K-I, K-C E-C M-C
Within Groups	1786.26	505	3.53			

Total	2036.27	509		I-C

According to Table 4.40 and Table 4.41 the F value for item 19 is [F(4,505)= 17.6, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.22, S.D.=1.76), is similar to the one from Electrical and Electronic students (Mean=5.05, S.D.=1.51) also it has a detectable difference with three departments including Mechanical Engineering (Mean=4.49, S.D.=1.82), Industrial Engineering (Mean=4.61, S.D.=1.88) and Civil Engineering (Mean=3.45, S.D.=2.11). Besides, the mean score of Electrical and Electronic students is similar to both means of Mechanical Engineering and Industrial Engineering, yet there exists a significant difference between the mean of Electrical and Electronic Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has a notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.42: Descriptive statistics for completing a program while having the language reference manual regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.75	1.84
Electrical and Electronic Engineering	60	4.86	1.75
Mechanical Engineering	81	4.24	1.86
Industrial Engineering	93	4.11	1.77
Civil Engineering	151	3.39	2.04
Total	510	4.16	1.96

Table 4.43: ANOVA summery for completing a program while having the language reference manual regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	163.88	4	40.97	11.50	0.01	K-I, K-C
Within Groups	1798.95	505	3.56			E-I, E-C
Total	1962.83	509				M-C I-C

According to table 4.42 and table 4.43 the F value for item 20 is [F(4,505)=11.50, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.75, S.D.=1.84), is similar to the one from Mechanical Engineering (Mean=4.24, S.D.=1.86), also it has a detectable difference with two departments including Industrial Engineering (Mean=4.11, S.D.=1.77) and Civil Engineering (Mean=3.39, S.D.= 2.046). Besides, the mean score of Electrical and Electronic students (Mean=4.86, S.D.=1.75), is similar to both means of Computer Engineering Mechanical Engineering, yet it differs considerably from the mean of two departments which are Industrial and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has an expressively difference with Civil Engineering.

Table 4.44: Descriptive statistics for completing a program by calling for help regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.16	1.82
Electrical and Electronic Engineering	60	5.06	1.74
Mechanical Engineering	81	4.50	2.06
Industrial Engineering	93	4.47	1.86

Civil Engineering	151	3.43	2.11
Total	510	4.40	2.06

Table 4.45: ANOVA summery for completing a program by calling for help regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	240.23	4	60.05	15.72	0.01	K-M, K-I, K-C
Within Groups	1929.11	505	3.82			E-C
Total	2169.35	509				M-C I-C

According to Table 4.44 and Table 4.45 the F value for item 21 is [F(4,505)=15.72, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.16, S.D.=1.82), is similar to the one from Electrical and Electronic students (Mean=5.06, S.D.=1.74) also it has a detectable difference with three departments including Mechanical Engineering (Mean=4.50, S.D.=2.06), Industrial Engineering (Mean=4.47, S.D.=1.86) and Civil Engineering (Mean=3.43, S.D.=2.11). On one hand, the mean score of Electrical and Electronic students is similar to the one from both Mechanical and Industrial Engineering, on the other it has an observable difference with Civil Engineering department. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. Then again, Industrial Engineering mean has difference comparing to Civil Engineering.

Table 4.46: Descriptive statistics for completing a program while starting with help regarding the department of engineering students

Department	N	Mean	Std. Deviation

Computer Engineering	125	5.16	1.87
Electrical and Electronic Engineering	60	4.98	1.63
Mechanical Engineering	81	4.40	1.87
Industrial Engineering	93	4.32	1.93
Civil Engineering	151	3.53	2.12
Total	510	4.38	2.03

Table 4.47: ANOVA summery for completing a program while starting with help regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	207.24	4	51.81	13.80	0.01	K-M, K-I, K-C E-I, E-C M-C I-C
Within Groups	1895.88	505	3.75			
Total	2103.12	509				

According to Table 4.46 and Table 4.47 the F value for item 22 is [F(4,505)=13.80, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.16, S.D.=1.87), is similar to the one from Electrical and Electronic students (Mean=4.98, S.D.=1.63) also it has a detectable difference with three departments including Mechanical Engineering (Mean=4.40, S.D.=1.87), Industrial Engineering (Mean=4.32, S.D.=1.93) and Civil Engineering (Mean=3.53, S.D.=2.12). Besides, the mean score of Electrical and Electronic students is similar to the one from Mechanical Engineering, on the contrary it has an observable difference with other two departments which are Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has notable difference comparing to Civil Engineering.

Table 4.48: Descriptive statistics for completing a program while having a lot of time regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.38	1.83
Electrical and Electronic Engineering	60	5.35	1.66
Mechanical Engineering	81	4.55	1.85
Industrial Engineering	93	4.33	1.93
Civil Engineering	151	3.26	2.12
Total	510	4.42	2.10

Table 4.49: ANOVA summery for completing a program while having a lot of time regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	371.67	4	92.91	24.96	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1879.28	505	3.72			
Total	2250.95	509				

According to Table 4.48 and Table 4.49 the F value for item 23 is [F(4,505)=24.96, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.38, S.D.=1.8), is similar to the one from Electrical and Electronic students (Mean=5.35, S.D.=1.66) also it has a detectable difference with three departments including Mechanical Engineering (Mean=4.55, S.D.=1.85), Industrial Engineering (Mean=4.33, S.D.=1.93) and Civil Engineering (Mean=3.26, S.D.=2.12). Additionally, the mean score of Electrical and Electronic Engineering students has a detectable difference with three departments including Mechanical Engineering, Industrial Engineering and Civil Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has notable difference comparing to Civil Engineering.

Table 4.50: Descriptive statistics for completing a program while having the built-in help facility regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.92	1.93
Electrical and Electronic Engineering	60	5.25	1.52
Mechanical Engineering	81	4.14	1.96
Industrial Engineering	93	4.07	1.68
Civil Engineering	151	2.98	1.77
Total	510	4.11	1.98

Table 4.51: ANOVA summery for completing a program while having the built-in help facility regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	352.35	4	88.09	26.98	0.01	K-M, K-I, K-C
Within Groups	1648.27	505	3.26			E-M, E-I, E-C
Total	2000.62	509				M-C I-C

According to Table 4.50 and Table 4.51 the F value for item 24 is [F(4,505)=26.98, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.92, S.D.=1.93), has a detectable difference with three departments including Mechanical Engineering (Mean=4.14, S.D.=1.96), Industrial Engineering (Mean=4.07, S.D.=1.68) and Civil Engineering (Mean=2.98, S.D.=1.77). Besides, the mean score of Electrical and Electronic students (Mean=5.25, S.D.=1.52) is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil

Engineering. Furthermore, the mean score of Mechanical Engineering, is similar to the one from Industrial Engineering department but it differs suggestively from the mean of Civil Engineering department. On the other hand, Industrial Engineering mean has notable difference comparing to Civil Engineering.

Table 4.52: Descriptive statistics for overcoming the problems while programming regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	5.03	1.86
Electrical and Electronic Engineering	60	5.23	1.65
Mechanical Engineering	81	4.20	1.86
Industrial Engineering	93	4.29	1.82
Civil Engineering	151	2.96	1.86
Total	510	4.17	2.01

Table 4.53: ANOVA summery for overcoming the problems while programming regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	383.15	4	95.78	28.57	0.01	K-M, K-I, K-C
Within Groups	1692.96	505	3.35			E-M, E-I, E-C
Total	2076.11	509				M-C I-C

According to Table 4.52 and Table 4.53 the F value for item 25 is [F(4,505)=28.57, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=5.03, S.D.=1.86), has a detectable difference with three departments including Mechanical Engineering (Mean=4.20, S.D.=1.86), Industrial Engineering (Mean=4.29, S.D.=1.82) and Civil Engineering (Mean=2.96, S.D.=1.86). Besides, the mean score of Electrical and Electronic students (Mean=5.23, S.D.=1.65) is similar to the one from Computer

Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.54: Descriptive statistics for coming up with a suitable strategy for a project regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.61	1.83
Electrical and Electronic Engineering	60	5.08	1.53
Mechanical Engineering	81	3.83	2.10
Industrial Engineering	93	3.96	1.78
Civil Engineering	151	2.58	1.62
Total	510	3.82	1.98

Table 4.55: ANOVA summery for coming up with a suitable strategy for a project regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	408.13	4	102.03	32.23	0.01	K-M, K-I, K-C E-M, E-I, E-C M-C I-C
Within Groups	1598.68	505	3.16			
Total	2006.81	509				

According to Table 4.54 and Table 4.55 the F value for item 26 is [F(4,505)=32,23, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.61, S.D.=1.83), has a detectable difference with three departments including Mechanical Engineering (Mean=3.83, S.D.=2.10), Industrial Engineering (Mean=3.96, S.D.=1.78) and Civil Engineering (Mean=2.58, S.D.=1.62). Besides, the mean score of Electrical and Electronic students (Mean=5.08, S.D.=1.53) is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.56: Descriptive statistics for managing the time while programming regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.56	1.96
Electrical and Electronic Engineering	60	5.06	1.51
Mechanical Engineering	81	4.09	1.95
Industrial Engineering	93	4.15	1.99
Civil Engineering	151	3.07	1.95
Total	510	4.03	2.03

Table 4.57: ANOVA summery for managing the time while programming regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	239.66	4	59.91	16.23	0.01	K-C
Within Groups	1863.83	505	3.69			E-M, E-I, E-C
Total	2103.49	509				M-C I-C

According to Table 4.56 and Table 4.57 the F value for item 27 is [F(4,505)=16.23, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.56, S.D.=1.96), is similar to the means of both Mechanical Engineering department

(Mean=4.09, S.D.=1.95) and Industrial Engineering department (Mean=4.15, S.D.=1.99), yet it has a major difference with the mean of Civil Engineering (Mean=3.07, S.D.=1.95). Besides, the mean score of Electrical and Electronic students (Mean=5.06, S.D.=1.51), is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.58: Descriptive statistics for tracing the implementation of a multi-file program regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.69	1.98
Electrical and Electronic Engineering	60	4.45	1.78
Mechanical Engineering	81	3.81	1.93
Industrial Engineering	93	3.93	1.78
Civil Engineering	151	2.33	1.65
Total	510	3.68	2.04

Table 4.59: ANOVA summery for tracing the implementation of a multi-file program regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	446.85	4	111.71	33.60	0.01	K-M, K-I, K-C
Within Groups	1678.57	505	3.32			E-M, E-C
Total	2125.42	509				M-C I-C

According to Table 4.58 and Table 4.59 the F value for item 28 is [F(4,505)= 33.60, p=0.01]. Additionally, the mean score of Computer Engineering (Mean=4.69, S.D.=1.98), is similar to the one from Electrical and Electronic students (Mean=4.45, S.D.=1.78) also it has a detectable difference with three departments including Mechanical Engineering (Mean=3.81, S.D.=1.93), Industrial Engineering (Mean=3.93, S.D.= 1.78), and Civil Engineering (Mean=2.33, S.D.=1.65). On one hand, the mean score of Electrical and Electronic students is similar to the one from Industrial Engineering, on the other it has an observable difference with both Mechanical and Civil Engineering departments. Then again, Mechanical Engineering mean has difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.60: Descriptive statistics for rewriting a part of a code to make it clear regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.66	1.91
Electrical and Electronic Engineering	60	4.88	1.65
Mechanical Engineering	81	3.81	1.98
Industrial Engineering	93	3.81	1.89
Civil Engineering	151	2.38	1.70
Total	510	3.72	2.06

Table 4.61: ANOVA summery for rewriting a part of a code to make it clear regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	463.66	4	115.91	34.35	0.01	K-M, K-I, K-C
Within Groups	1703.90	505	3.37			E-M, E-I, E-C
Total	2167.56	509				M-C , I-C

According to Table 4.60 and Table 4.61 the F value for item 29 is [F(4,505)=34.35, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.66, S.D.= 1.91), has a detectable difference with three departments including Mechanical Engineering (Mean=3.81, S.D.=1.98), Industrial Engineering (Mean=3.81, S.D.=1.89), and Civil Engineering (Mean=2.38, S.D.=1.70). Besides, the mean score of Electrical and Electronic students (Mean=4.88, S.D.=1.65), is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.62: Descriptive statistics for concentrating on programing regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.62	1.92
Electrical and Electronic Engineering	60	4.91	1.53
Mechanical Engineering	81	3.95	1.90
Industrial Engineering	93	4.11	1.92
Civil Engineering	151	2.91	1.97
Total	510	3.95	2.02

Table 4.63: ANOVA summery for concentrating on programing regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	277.57	4	69.39	19.32	0.01	K-M, K-C E-M, E-I, E-C
Within Groups	1813.29	505	3.59			

Total	2090.87	509			M-C I-C

According to Table 4.62 and Table 4.63 the F value for item 30 is [F(4,505)=19.32, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.62, S.D.=1,92), has a detectable difference with three departments including Mechanical Engineering (Mean=3.95, S.D.=1.90), Industrial Engineering (Mean=4.11, S.D.=1.92), and Civil Engineering (Mean=2.91, S.D.=1.97). Besides, the mean score of Electrical and Electronic students (Mean=4.91, S.D.=1.53), is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.64: Descriptive statistics for self-motivation to program regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.57	1.79
Electrical and Electronic Engineering	60	4.90	1.59
Mechanical Engineering	81	3.83	1.92
Industrial Engineering	93	4.06	1.92
Civil Engineering	151	3.07	2.12
Total	510	3.96	2.02

Table 4.65: ANOVA summery for self-motivation to program regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	219.71	4	54.92	14.88	0.01	K-M, K-C E-M, E-I, E-C M-C I-C
Within Groups	1863.50	505	3.69			
Total	2083.21	509				

Considering Table 4.64 and Table 4.65 the F value for item 31 is [F(4,505)=14.88, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.57, S.D.=1.79), has a detectable difference with three departments including Mechanical Engineering (Mean=4.06, S.D.=1.92), Industrial Engineering (Mean=4.06, S.D.=1.92), and Civil Engineering (Mean=3.07, S.D.=2.12). Besides, the mean score of Electrical and Electronic students (Mean=4.90, S.D.=1.59), is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Table 4.66: Descriptive statistics for writing an understandable program regarding the department of engineering students

Department	N	Mean	Std. Deviation
Computer Engineering	125	4.88	1.86
Electrical and Electronic Engineering	60	5.05	1.59
Mechanical Engineering	81	3.79	2.05
Industrial Engineering	93	3.89	1.79
Civil Engineering	151	2.70	1.97
Total	510	3.90	2.08

Table 4.67: ANOVA summery for writing an understandable program regarding the department of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	417.10	4	104.27	29.28	0.01	K-M, K-I, K-C
Within Groups	1797.99	505	3.56			E-M, E-I, E-C
Total	2215.09	509				M-C I-C

As stated in Table 4.66 and Table 4.67, it can be said that the F value for item 32 is [F(4,505)=29.28, p=0.01]. Additionally, the mean score of Computer Engineering students (Mean=4.88, S.D.=1.86), has a noticeable difference with three departments including Mechanical Engineering (Mean=3.79, S.D.=2.05), Industrial Engineering (Mean=3.89, S.D.=1.79), and Civil Engineering (Mean=2.70, S.D.=1.97). Besides, the mean score of Electrical and Electronic students (Mean=5.05, S.D.=1.59), is similar to the one from Computer Engineering, on the contrary it has an observable difference with other three departments which are Mechanical Engineering, Industrial Engineering and Civil Engineering. On the other hand, Mechanical Engineering mean has notable difference comparing to Civil Engineering. Furthermore, the mean score of Industrial Engineering, is similar to the one from Mechanical Engineering department but it differs suggestively from the mean of Civil Engineering department.

Therefore, in 16 Items out of 32, Computer Engineering students had the highest mean value and for the other 50% of the Items Electrical and Electronic students had the highest mean. Thus this research cannot approve that Computer Engineering student have a higher CPSE level than Electrical and Electronics engineering students which is what Korkmaz & Altun (2014) concluded. On the other hand Askar & Davenport

(2009), stated that Computer Engineering students' self-efficacy scores were considerably higher than that of students from the other engineering departments.

4.4 CPSE level of engineering students regarding class level

For figuring out the connection between Computer Programming Self-Efficacy and class level, a One Way ANOVA test and a Post Hoc comparison were used to evaluate the impacts of grade (1st Year,2nd Year, 3rd Year, 4th Year) on Computer Programming Self-Efficacy. As a result, the ANOVA table displayed that there is no noteworthy difference in most items except in 10 items regarding the students class level.

Table 4.68: Descriptive statistics for writing a program that displays a greeting message regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	4.82	1.91
2nd Year	131	5.02	2.03
3rd Year	179	4.44	2.18
4th Year	131	4.93	2.17
Total	510	4.76	2.11

Table 4.69: ANOVA summery for writing a program that displays a greeting message regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	31.33	3	10.44	2.34	0.07	2nd-3rd 4th-3rd
Within Groups	2253.36	506	4.45			
Total	2284.69	509				

According to Table 4.68 and Table 4.69 and post hoc comparison for item 4 it can be concluded that the mean value for 1st Year students (Mean=4.82, S.D.= 1.91) is similar to the one from 3rd Years (Mean=4.44, S.D.=2.18). Besides the mean value of 2nd Year students (Mean=5.02, S.D.= 2.03) is similar to both 1st Year and 4th Year students (Mean=4.93, S.D.=2.17), also it has a significant difference with the mean of 3rd Year students. Furthermore, the mean for and 4th Year students is similar to 1st Year students but it differs meaningfully from 3rd Year students.

Table 4.70: Descriptive statistics for writing a small program according to a small familiar problem regarding grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	4.17	1.96
2nd Year	131	3.90	1.93
3rd Year	179	4.17	2.09
4th Year	131	4.48	2.27
Total	510	4.18	2.08

Table 4.71: ANOVA summery for writing a small program according to a small familiar problem regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	22.68	3	7.56	1.74	0.15	4th-2nd
Within Groups	2197.98	506	4.34			
Total	2220.67	509				

As shown on Table 4.70, Table 4.71 and post hoc comparison for item 9 it can be concluded that the mean value of 1st Year students (Mean=4.17, S.D.=1.96) is similar to both 2nd Year (Mean=3.90, S.D.=1.93) and 3rd Year students (Mean=4.17, S.D.= 2.09). Also 3rd Year students mean is similar to the one from 2nd Year students. Furthermore, the mean for and 4th Year students is similar to both 1st Year and 3rd Year students but it differs suggestively from 2nd Year students.

Table 4.72: Descriptive statistics for writing a program while the specifications are defined regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.30	1.80
2nd Year	131	3.15	1.89
3rd Year	179	3.67	2.04
4th Year	131	3.74	2.32
Total	510	3.50	2.06

Table 4.73: ANOVA summery for writing a program while the specifications are defined regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	31.19	3	10.39	2.46	0.06	3^{rd}-2^{nd} 4^{th}-2^{nd}
Within Groups	2138.28	506	4.22			
Total	2169.48	509				

As shown on Table 4.72, Table 4.73 and post hoc comparison for item 11 it can be concluded that the mean value for 1^{st} Year students (Mean=3.30, S.D.=1.80) is similar to the one from 2^{nd} Years (Mean=3.15, S.D.=1.89). Furthermore, the mean for and 3^{rd} Year students is similar to 1^{st} Year students but it differs meaningfully from 2^{nd} Year students. Besides the mean value of 4^{th} Year students (Mean=3.74, S.D.=2.32) is similar to both 1^{st} Year and 3^{rd} Year students (Mean=3.67, S.D.=2.04), also it has a significant difference with the mean of 2^{nd} Year students.

Table 4.74: Descriptive statistics for organizing and designing a program in a modular regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.56	1.76
2nd Year	131	3.16	1.85
3rd Year	179	3.74	2.10
4th Year	131	3.93	2.31

Total	510	3.61	2. 07

Table 4.75: ANOVA summery for organizing and designing a program in a modular regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	43.54	3	14.51	3.42	0.01	3^{rd}-2^{nd} 4^{th} -2^{nd}
Within Groups	2146.65	506	4.24			
Total	2190.20	509				

Table 4.74 and table 75 illustrate that for item 12 there exists an important difference and there is a correlation between student's class level and their CPSE level. Besides the mean score for 1^{st} Year students (Mean=3.56, S.D.=1.76) is close to the one for 2^{nd} Year students (Mean=3.16, S.D.=1.85). Also the mean score for 3^{rd} Year students (Mean=3.74, S.D.=2.103) is similar to 1^{st} Year students (Mean=3.56, S.D.=1.76) and has an important difference with 2^{nd} Year students (Mean=3.16, S.D.=1.85). The mean score for 4^{th} Year students (Mean=3.93, S.D.=2.31) is similar to the mean of both 1^{st} Year students (Mean=3.56, S.D.=1.76) and 3^{rd} Year students (Mean=3.74, S.D.=2.10). Yet it differs significantly from the mean of 2^{nd} Year students (Mean=3.16, S.D.=1.85).

Table 4.76: Descriptive statistics for understanding the object-oriented paradigm regarding the class level of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.37	1.72
2nd Year	131	3.22	1.98
3rd Year	179	3.68	2.12
4th Year	131	4.05	2.19
Total	510	3.62	2.07

Table 4.77: ANOVA summery for understanding the object-oriented paradigm regarding the class level of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	49.52	3	16.50	3.89	0.01	4^{th} -1^{st}
Within Groups	2142.43	506	4.23			4^{th} -2^{nd}
Total	2191.96	509				

According to table 4.76 and table 4.77 and post hoc comparison for item 13 it can be concluded that the mean value for 1^{st} Year students (Mean=3.37, S.D.=1.72) is similar to the one for 2^{nd} Year students (Mean=3.22, S.D.=1.98). Furthermore, the mean for and 3^{rd} Year students (Mean=3.68, S.D.=2.120) is similar to both 1^{st} Year students (Mean=3.37, S.D.=1.72) and 2^{nd} Year students (Mean=3.22, S.D.=1.98). The mean score for 4^{th} Year students (Mean=4.05, S.D.=2.19) is similar to the mean of 3^{rd} Year students (Mean=3.68, S.D.=2.12) yet it differs expressively with the mean of 2^{nd} Year students (Mean=3.22, S.D.=1.98) and 1^{st} Year students (Mean=3.37, S.D.=1.72).

Table 4.78: Descriptive statistics for identifying and defining the objects in the problem domain regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.79	1.82
2nd Year	131	3.46	1.96
3rd Year	179	3.84	2.17
4th Year	131	4.11	2.11
Total	510	3.80	2.06

Table 4.79: ANOVA summery for identifying and defining the objects in the problem domain regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	27.89	3	9.29	2.19	0.08	4^{th}-2^{nd}
Within Groups	2148.65	506	4.24			

Total	2176.55	509				

According to Table 4.78, Table 4.79 and post hoc comparison for item 14 it can be concluded that the mean value for 1st Year students (Mean=3.79, S.D.=1.82) is similar to the one from 2nd Years (Mean=3.46, S.D.=1.96). Besides the mean value of 2nd Year students is similar to both 1st Year and 3rd Year students (Mean=3.84, S.D.=2.17). On the other hand, the mean value of 4th Year students (Mean=4.11, S.D.=2.11) is similar to both 1st Year and 3rd Year students, also it has a significant difference with the mean of 2nd Year students.

Table 4.80: Descriptive statistics for making use of a pre-written function regarding the class level of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.89	1.82
2nd Year	131	3.64	1.97
3rd Year	179	4.04	2.10
4th Year	131	4.36	2.29
Total	510	4.00	2.09

Table 4.81: ANOVA summery for making use of a pre-written function regarding the class level of engineering students

Variable Source	Sum of Squares	Df	Mean Square	F	Sig.	Significant Difference
Between Groups	34.79	3	11.59	2.66	.048	4th-2nd
Within Groups	2206.19	506	4.36			
Total	2240.98	509				

Table 4.80 and Table 4.81 suggest that the mean score of 1st Year students (Mean=3.89, S.D.=1.82) is close to the mean number of 2nd Year students (Mean=3.64,

S.D.=1.97). On the other hand, the mean of 3rd Year students (Mean=4.04, S.D.=2.10) is similar to both 1st Year students and 2nd Year students mean number. Additionally, the mean score for 4th Year students (Mean=4.36, S.D.=2.29) has a substantial difference with the 2nd Year students mean nevertheless it is similar to the means of 1st Year students and 3rd Year students.

Table 4.82: Descriptive statistics for using a defined class while having a labeled declaration of it regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.62	1.99
2nd Year	131	3.43	1.89
3rd Year	179	3.92	2.10
4th Year	131	4.07	2.30
Total	510	3.79	2.10

Table 4.83: ANOVA summery for using a defined class while having a labeled declaration of it regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	32.50	3	10.83	2.47	0.06	3rd-2nd 4th-2nd
Within Groups	2215.69	506	4.37			
Total	2248.19	509				

As shown on Table 4.82, Table 4.83 and post hoc comparison for item 16 it can be concluded that the mean value for 1st Year students (Mean=3.62, S.D.=1.99) is similar to the one from 2nd Years (Mean=3.43, S.D.=1.89). Furthermore, the mean for and 3rd Year students is similar to 1st Year students but it differs meaningfully from 2nd Year students. On the other hand, the mean value of 4th Year students (Mean=4.07, S.D.=2.30) is similar to both 1st Year and 3rd Year (Mean=3.92, S.D.=2.10), students, also it has a significant difference with the mean of 2nd Year students.

Table 4.84: Descriptive statistics for completing a program while having the language reference manual regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.75	1.85
2nd Year	131	3.99	1.96
3rd Year	179	4.33	1.90
4th Year	131	4.32	2.07
Total	510	4.16	1.96

Table 4.85: ANOVA summery for completing a program while having the language reference manual regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	24.25	3	8.08	2.11	0.09	3rd-1st 4th-1st
Within Groups	1938.57	506	3.83			
Total	1962.83	509				

As shown on Table 4.84, Table 4.85 and post hoc comparison for item 20 it can be concluded that the mean value for 2nd Year students (Mean=3.99, S.D.=1.96) is similar to the one from 1st Years (Mean=3.75, S.D.=1.85). Besides the mean value of 3rd Year students (Mean=4.33, S.D.=1.90) is similar to both 2nd Year and 4th Year students (Mean=4.32, S.D.=2.07), also it has a significant difference with the mean of 1st Year students. Furthermore, the mean for and 4th Year students is similar to 2nd Year students but it differs meaningfully from 1st Year students.

Table 4.86: Descriptive statistics for overcoming the problems while programming regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	4.04	1.89
2nd Year	131	4.00	1.94
3rd Year	179	4.10	2.05
4th Year	131	4.52	2.08

Total	510	4.17	2.01

Table 4.87: ANOVA summery for overcoming the problems while programming regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference
Between Groups	22.40	3	7.46	1.84	0.13	4th-2nd
Within Groups	2053.71	506	4.05			
Total	2076.11	509				

As shown on Table 4.86, Table 4.87 and post hoc comparison for item 25 it can be concluded that the mean value for 1st Year students (Mean=4.04, S.D.=1.89) is similar to the one from 2nd Years (Mean=4.00, S.D.= 1.94). Besides the mean value of 3rd Year students (Mean=4.10, S.D.=2.05) is similar to both 1st Year and 2nd Year students mean. On the other hand, the mean value of 4th Year students (Mean=4.52, S.D.=2.08) is similar to both 1st Year and 3rd Year students, also it has a significant difference with the mean of 2nd Year students.

Table 4.88: Descriptive statistics for coming up with a suitable strategy for a project regarding the grade of engineering students

Class level	N	Mean	Std. Deviation
1st Year	69	3.59	1.88
2nd Year	131	3.56	1.86
3rd Year	179	3.87	2.01
4th Year	131	4.15	2.07
Total	510	3.82	1.98

Table 4.89: ANOVA summery for coming up with a suitable strategy for a project regarding the grade of engineering students

Variable Source	Sum of Squares	df	Mean Square	F	Sig.	Significant Difference

Between Groups	26.98	3	8.99	2.29	0.07	4^{th}-2^{nd}
Within Groups	1979.82	506	3.91			
Total	2006.81	509				

By paying attention to Table 4.88, Table 4.89 and post hoc comparison for item 26 it can be said that the mean value for 1^{st} Year students (Mean=3.59, S.D.=1.88) is similar to the one from 2^{nd} Years (Mean=3.56, S.D.=1.86). Besides the mean value of 3^{rd} Year students (Mean=3.87, S.D.=2.01) is similar to both 1^{st} Year and 2^{nd} Year students. On the other hand, the mean value of 4^{th} Year students (Mean=4.15, S.D.=2.07) is similar to both 1^{st} Year and 3^{rd} Year students, also it has a significant difference with the mean of 2^{nd} Year students.

Hence for 65% of the Items there was no substantial difference among the class levels but concerning the Items which there was a difference among grades of students, 4^{th} Year students had the highest mean value and 2^{nd} Year students had the lowest. It can be said that the pre-knowledge of computer programming can be an important factor influencing the CPSE level of 1^{st} and 2^{nd} Year students, besides since all engineering students should take programming courses, it was expected to see that 4^{th} Year students had the highest mean. However in a study by Ozyurt (2015), second year students had higher CPSE level.

Chapter 5

CONCLUSION

This study was about Computer Programming Self-Efficacy (CPSE) level of students. The research method of this study was quantitative method, using the survey technique and the participants were from 5 different engineering department at Eastern Mediterranean University (EMU). According to the findings, the CPSE level of engineering bachelor students registered at for fall 2018-2019, is at an average level.

For this research, there were 105 female and 405 male volunteers and the results showed that computer programming self-efficacy level of females were higher than males in most tasks.

The findings displayed that students studying Civil Engineering are mostly not confident in computer programming tasks, however, volunteers from computer Engineering department and electrical and electronic department were fairly confident and had a higher self-efficacy level, while students from Mechanical Engineering department and Industrial Engineering department were slightly confident regarding computer programming tasks and had a higher self-efficacy comparing to people from civil engineering department.

Results proved that the computer programming self-efficacy level is different according to the class level of the students. Regarding the grade of the participants in this study who were 1st Year, 2nd Year, 3rd Year, and 4th Year students; as it was expected and not unusual, computer programming self-efficacy level of 4th Year students were higher than others.

REFERENCES

Anastasiadou, S. D., & Karakos, A. S. (2011). The beliefs of Electrical and Computer Engineering students' regarding computer programming. *International Journal of Technology, Knowledge and Society*, *7*(1), 37–51. https://doi.org/10.18848/1832-3669/CGP/v07i01/56170

Askar, P., & Davenport, D. (2009). An investigation of factors related to self-efficacy for java programming among engineering students. *Turkish Online Journal of Educational Technology*. https://doi.org/10.1186/1476-069X-10-98

Bandura, A. (1986). *Social foundations of thought and action: Social cognitive theory. Englewood Cliffs, NJ: Prentice Hall*. *Englewood Cliffs, New Jersey: Prentice Hall*. New Jersey.

Bandura, A. (1989). *Social cognitive theory. In R. Vasta (Ed.). Annals of child development. Vol. 6. Six theories of child development* (Vol. 6). JAI Press. https://doi.org/10.1111/1467-839X.00024

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

Bandura, A. (2006). BanduraGuide2006. *Guide for Constructig Self-Efficacy Scales*, 307–337. https://doi.org/10.1017/CBO9781107415324.004

Beckers, J. J., & Schmidt, H. G. (2001). Structure of computer anxiety: A six-factor model. *Computers in Human Behavior*, *17*(1), 35–49. https://doi.org/10.1016/S0747-5632(00)00036-4

Bell, D. (2005). *Software Engineering for Students: A Programming Approach*. *Engineering*. Retrieved from www.pearsoned.co.uk

Berland, M., & Lee, V. (2011). Collaborative strategic board games as a site for distributed computational thinking. *Other Computer Sciences*. https://doi.org/10.4018/ijgbl.2011040105

Bilge, F., Tuzgöl Dost, M., & Çetin, B. (2014). Factors affecting burnout and school engagement among high school students: Study habits, self-efficacy beliefs, and academic success. *Kuram ve Uygulamada Egitim Bilimleri*, *14*(5), 1721–1727. https://doi.org/10.12738/estp.2014.5.1727

Binkley, M., Erstad, O., Herman, J., Raizen, S., Ripley, M., Miller-Ricci, M., & Rumble, M. (2014). Defining twenty-first century skills. In *Assessment and teaching of 21st century skills* (pp. 17–66). Dordrecht: Springer Netherlands. https://doi.org/10.1007/978-94-007-2324-5_2

Bonett, D. G., & Wright, T. A. (2015). Cronbach's alpha reliability: Interval estimation, hypothesis testing, and sample size planning. *Journal of Organizational Behavior*, *36*(1), 3–15. https://doi.org/10.1002/job.1960

Borgman, C., Abelson, H., Dirks, L., Johnson, R., Koedinger, K. R., Linn, M. C., … Others. (2008). *Fostering learning in the networked world: The cyberlearning* *opportunity and challenge*. *Washington, DC: NSF*. Retrieved from https://www.nsf.gov/pubs/2008/nsf08204/nsf08204.pdf

Brito, M., & De-Sá-Soares, F. (2014). Assessment frequency in introductory computer programming disciplines. *Computers in Human Behavior*, *30*, 623–628. https://doi.org/10.1016/j.chb.2013.07.044

Cramer, D., & Howitt, D. (2004). *The Sage dictionary of statistics : a practical resource for students in the social sciences*. *Published in* ***2004*** *in London by Sage*. https://doi.org/10.5194/bg-12-4361-2015

Dagiene, V. ., Skupas, B. ., & Kurilovas, E. a. . (2014). Programming assignments in virtual learning environments: Developments and opportunities for engineering education. *International Journal of Engineering Education*.

Dao, V. H. (2018). *TECHNOLOGY AND ELDERLY WELL-BEING*. Retrieved from https://search.proquest.com/docview/2089495966?pq-origsite=summon

Davidsson, K., Larzon, L.-åke, & Ljunggren, K. (2013). Self-Efficacy in Programming among STS Students. Retrieved from http://www.it.uu.se/edu/course/homepage/datadidaktik/ht10/reports/Self-Efficacy.pdf

Erdogan, Y., Aydin, E., & Kabaca, T. (2008). Exploring the psychological predictors of programming achievement. *Journal of Instructional Psychology*.

Fang, X. (2012). Application of the participatory method to the computer fundamentals course. In *Advances in Intelligent and Soft Computing* (Vol. 137 AISC, pp. 185–189). Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-27866-2_23

Garner, S. (2009). Learning to program from scratch. In *Proceedings - 2009 9th IEEE International Conference on Advanced Learning Technologies, ICALT 2009* (pp. 451–452). IEEE. https://doi.org/10.1109/ICALT.2009.50

Gencturk, A., & Korucu, A. (2017). The Effects of Web 2.0 Technologies Usage in Programming Languages Lesson on the Academic Success, Interrogative Learning Skills and Attitudes of Students towards Programming Languages. *Higher Education Studies*, *7*(1), 114. https://doi.org/10.5539/hes.v7n1p114

Given, L. . (2008). *the sage encyclopedia of qualitative research methods, volumes 1&2*. *Context*. https://doi.org/http://dx.doi.org/10.4135/9781412963909

Gökçearslan, Ş., Günbatar, M. S., & Kukul, V. (2017). COMPUTER PROGRAMMING SELF-EFFICACY SCALE (CPSES) FOR SECONDARY SCHOOL STUDENTS: DEVELOPMENT, VALIDATION AND RELIABILITY. *Eğitim Teknolojisi Kuram ve Uygulama*, *7*(1), 158–158. https://doi.org/10.17943/etku.288493

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

Govender, I., Govender, D., Havenga, M., Mentz, E., Breed, B., Dignum, F., & Dignum, V. (2014). Increasing self-efficacy in learning to program: exploring the benefits of explicit instruction for problem solving. *TD The Journal for Transdisciplinary Research in Southern Africa*, *10*(1), 187–200. Retrieved from https://search.proquest.com/docview/1898638659/fulltextPDF/C4B47FC0BDC14B9DPQ/1?accountid=15792

Graham, R. (2012). *Achieving excellence in engineering education: the ingredients of successful change*. *The Royal Academy of Engineering* (Vol. 101). Retrieved from http://www.rhgraham.org/resources/Educational-change-summary-report.pdf

Heale, R., & Twycross, A. (2015). Validity and reliability in quantitative studies. *Evidence Based Nursing*, *18*(3), 66–67. https://doi.org/10.1136/eb-2015-102129

Hilal, M. M., Suzastri, A. A., & Zulhazlin, A. (2018). Computer Self-Efficacy in Programming Language for Non-Technical Students using Scratch Programming. *International Journal of Pure and Applied Mathematics*, *118*. Retrieved from http://scratch.mit.edu

Hughes, T. P. (2004). Human-built world: How to think about technology and culture. *Scienceculture*, xii, 223 p. https://doi.org/10.1109/MTAS.2006.1649029

IEEEComputerSociety. (2004). *computer Engineering Curricula*. Retrieved from http://delivery.acm.org/10.1145/2480000/2479869/ce-final-report-1.pdf?ip=193.140.42.143&id=2479869&acc=NO RULES&key=956257

EA1AE17323.01079F2FA8E70573.4D4702B0C3E38B35.4D4702B0C3E38B35&__acm__=1544917917_5a2c503dd0b4ce127a293fd15e2e84bf

Jackson, D., & Miller, R. (2009). A New Approach to Teaching Programming. https://doi.org/10.1111/j.1558-5646.2009.00939.x

Kafai, Y. B., & Burke, Q. (2013). Computer programming goes back to school: Learning programming introduces students to solving problems, designing applications, and making connections online. *Phi Delta Kappan*, *95*(1), 61–65. https://doi.org/10.1177/003172171309500111

Kalelioglu, F. (2015). A new way of teaching programming skills to K-12 students: Code.org. *Computers in Human Behavior*, *52*(C), 200–210. https://doi.org/10.1016/j.chb.2015.05.047

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming. *ACM Computing Surveys*, *37*(2), 83–137. https://doi.org/10.1145/1089733.1089734

Khorrami-Arani, O. (2001). *Researching computer self-efficacy THE CONCEPT AND MEASUREMENT OF SELF-EFFICACY*. *International Education Journal* (Vol. 2). Retrieved from http://www.flinders.edu.au/education/iej

Korkmaz, Ö., & Altun, H. (2014). Adapting Computer Programming Self-Efficacy Scale and Engineering Students' Self-Efficacy Perceptions. *Participatory Educational Research*, *1*(1), 20–31. https://doi.org/10.17275/per.14.02.1.1

Kutta-JR, T. J. (2017). *NOTE-TAKING AND TECHNOLOGY By TIMOTHY J . KUTTA JR . A Dissertation submitted to the Department of Psychology in partial fulfillment of the requirements for the degree of Doctor of Philosophy*.

Lachapelle, C. P., Cunningham, C. M., & Oh, Y. (2018). What is technology? Development and evaluation of a simple instrument for measuring children's conceptions of technology. *International Journal of Science Education*, 1–22. https://doi.org/10.1080/09500693.2018.1545101

Levy, F., & Murnane, R. J. (2004). *The new division of labor : how computers are creating the next job market*. Princeton University Press Princeton, NJ, USA: Russell Sage Foundation. Retrieved from https://dl.acm.org/citation.cfm?id=975438

Li, P. L., Ko, A. J., & Zhu, J. (2015). What makes a great software engineer? In *Proceedings - International Conference on Software Engineering* (Vol. 1, pp. 700–710). https://doi.org/10.1109/ICSE.2015.335

Maddrey, E. (2011). The Effect of Problem-Solving Instruction on the Programming Self- efficacy and Achievement of Introductory Computer Science Students. *Information Sciences*, 120. Retrieved from https://search.proquest.com/docview/884796506?pq-origsite=summon

Marakas, G., Yi, M., & Johnson, R. (1998). The Multilevel and Multifaceted Character of Computer Self-Efficacy: Toward Clarification of the Construct and an Integrative Framework for Research. *Information Systems Research*, *9*(2), 126–

163. https://doi.org/10.1287/isre.9.2.126

Murphy, C. A., Coover, D., & Owen, S. V. (1989). Development and Validation of the Computer Self-Efficacy Scale. *Educational and Psychological Measurement*, *49*(4), 893–899. https://doi.org/10.1016/j.physb.2008.08.008

Neuman, W. L. (2014). Social Research Methods: Qualitative and Quantitative Approaches. *Teaching Sociology*. https://doi.org/10.2307/3211488

Nilsen, H., & Larsen, E. A. (2011). Using the personalized system of instruction in an introductory programming course. In *NOKOBIT*.

Norrbom, M. (2018). *Self-Efficacy of School Psychologists*. Fairleigh Dickinson University,. Retrieved from https://search.proquest.com/docview/2112361028?pq-origsite=summon

Ozyurt, O. (2015). *An analysis on /distance Education Computer Programming Studetns Attitudes Regarding Programming and their Self-efficacy for Programming*. *Turkish Online Journal of Distance Education*. Retrieved from http://tojde.anadolu.edu.tr/yonetim/icerik/makaleler/1115-published.pdf

Pajares, F. (1997). Current directions in self-efficacy research. Advances in motivation and achievement. *Psychological Science*.

Patzek, T. W., & Juanes, R. (2006). *An Introduction to Computer Programming for Engineers and Scientists*. *Environmental Engineering*.

Ramalingam, V., LaBelle, D., & Wiedenbeck, S. (2004). Self-efficacy and mental models in learning to program. *ACM SIGCSE Bulletin*. https://doi.org/10.1145/1026487.1008042

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

Roberts, P., Priest, H., & Traynor, M. (2006). Reliability and validity in research. *Nursing Standard*. https://doi.org/10.7748/ns2006.07.20.44.41.c6560

Saunders, M., Lewis, P., & Thornhill, A. (2009). *Research methods for business students*. *Research methods for business students*. https://doi.org/10.1080/10810730.2015.1018568

Smith, D. C., Cypher, A., & Tesler, L. (2000). Programming by example: novice programming comes of age. *Communications of the ACM*, *43*(3), 75–81. https://doi.org/10.1145/330534.330544

Soykan, F., & Kanbul, S. (2018). Analysing k12 students' self-efficacy regarding coding education. *TEM Journal*, *7*(1), 182–187. https://doi.org/10.18421/TEM71-22

Tsai, M.-J., Wang, C.-Y., & Hsu, P.-F. (2018). Developing the Computer Programming Self-Efficacy Scale for Computer Literacy Education. *Journal of*

Educational Computing Research, *56*(8), 073563311774674. https://doi.org/10.1177/0735633117746747

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Yildiz Durak, H. (2018). Digital story design activities used for teaching programming effect on learning of programming concepts, programming self-efficacy, and participation and analysis of student experiences. *Journal of Computer Assisted Learning*, *34*(6), 740–752. https://doi.org/10.1111/jcal.12281

Zampirolli, F. A., Goya, D., Pimentel, E. P., & Kobayashi, G. (2018). Evaluation process for an introductory programming course using blended learning in engineering education. *Computer Applications in Engineering Education*, *26*(6), 2210–2222. https://doi.org/10.1002/cae.22029

APPENDICES

Appendix A (Participants demographic information)

Dear student, Please put a tick ✓ in the appropriate box which best suits the answer you have selected.

Part 1: Demographics

1. Gender:
   - ○ Female
   - ○ Male
2. Which department do you study at?
   - ○ Computer Engineering department
   - ○ Electrical & Electronic Engineering department
   - ○ Mechanical Engineering department
   - ○ Industrial Engineering department
   - ○ Civil Engineering department
3. What is your academic class level (grade)?
   - ○ 1^{st} year
   - ○ 2^{nd} year
   - ○ 3^{rd} year
   - ○ 4^{th} year

Appendix A (Questionnaire including 32-items)

Computer Programming Self-Efficacy Scale

Rate your confidence in doing the following computer programming related tasks using a scale of 1 (not at all confident) to 7 (absolutely confident). If a specific term or task is totally unfamiliar to you, please mark 1.

		1	2	3	4	5	6	7
1	I could write syntactically correct coding statements.							
2	I could understand the language structure of computer programming languages and the usage of the reserved words.							
3	I could write logically correct blocks of code using a computer program language.							
4	I could write a program that displays a greeting message.							
5	I could write a program that computes the average of three numbers.							
6	I could write a program that computes the average of any given number of numbers.							
7	I could use built-in functions that are available in the various programming applications.							
8	I could build my own application.							
9	I could write a small program given a small problem that is familiar to me.							
10	I could write a reasonably sized program that can solve a problem which is only unclearly familiar to me.							
11	I could write a long and complex program to solve any given problem as long as the specifications are clearly defined.							
12	I could organize and design my program in a modular manner.							
13	I could understand the object-oriented paradigm.							
14	I could identify the objects in the problem domain and could declare, define, and use them.							
15	I could make use of a pre-written function, given a clearly labeled declaration of the function.							
16	I could make use of a class that is already defined, given a clearly labeled declaration of the class.							
17	I could debug (correct all the errors) a long and complex program that I had written and make it work.							

18	I could comprehend a long, complex multi-file program.							
19	I could complete a programming project if someone showed me how to solve the problem first.							
20	I could complete a programming project if I had only the language reference manual for help.							
21	I could complete a programming project if I could call someone for help if I got stuck.							
22	I could complete a programming project once someone else helped me get started.							
23	I could complete a programming project if I had a lot of time to complete the program.							
24	I could complete a programming project if I had just the built-in help facility for assistance.							
25	While working on a programming project, if I got stuck at a point I could find ways of overcoming the problem.							
26	I could come up with a suitable strategy for a given programming project in a short time.							
27	I could manage my time efficiently if I had a pressing deadline on a programming project.							
28	I could mentally trace through the execution of a long, complex multi-file program given to me.							
29	I could rewrite lengthy and confusing portions of code to be more readable and clear.							
30	I could find a way to concentrate on my program, even when there were many distractors around me.							
31	I could find ways of motivating myself to program, even if the problem area was of no interest tome.							
32	I could write a program that someone else could comprehend and add features to at a later date.							

Appendix B: Ethics committee permission letter

Doğu Akdeniz Üniversitesi
"Erdem, Bilgi, Gelişim"

Eastern Mediterranean University
"Virtue, Knowledge, Advancement"

99628, Gazimağusa, KUZEY KIBRIS / Famagusta, North Cyprus, via Mersin-10 TURKEY
Tel: (+90) 392 630 1995
Faks/Fax: (+90) 392 630 2919
E-mail: bayek@emu.edu.tr

Etik Kurulu / *Ethics Committee*

Reference No: ETK00-2018-0288 12.11.2018
Subject: Application for Ethics.

RE: Golmehr Esmaeilzadeh Hanjani
Faculty of Education

To Whom It May Concern:

On the date of **12.11.2018**, (Meeting number ***2018/61-08),*** EMU's Scientific Research and Publication Ethics Committee (BAYEK) has granted**,** Golmehr Esmaeilzadeh Hanjani , from the, Faculty of Education to pursue with her MA. thesis work **"Computer Programming Self-Efficacy Levels (CPSES) of Engineering Student: An Example of Eastern Mediterranean University (EMU)"** under the supervision of Assoc. Prof. Dr. Ersun İşçioğlu. This decision has been taken by the majority of votes.

Regards,

Assoc. Prof. Dr. Şükrü Tüzmen
Director of Ethics Commitee

ŞT/ba.

www.emu.edu.tr

Appendix C: Turnitin report

Turnitin Originality Report

Turnitin Originality Report

Thesis by Golmehr H.

From Golmehr (SCHOOL OF COMPUTING AND TECHNOLOGY)

- Processed on 17-Jan-2019 12:35 +03
- ID: 1065171368
- Word Count: 23231

Similarity Index
15%
Similarity by Source

Internet Sources:
13%
Publications:
10%
Student Papers:
6%

sources:

1 2% match (student papers from 06-Jul-2018)
Submitted to International Islamic University Malaysia on 2018-07-06

2 1% match (Internet from 30-Jul-2013)
http://www.tojet.net/volumes/v8i1.pdf

3 1% match (publications)
Hatice Yildiz Durak. "Digital story design activities used for teaching programming effect on learning of programming concepts, programming self-efficacy, and participation and analysis of student experiences", Journal of Computer Assisted Learning, 2018

4 1% match (Internet from 03-Apr-2014)
http://perweb.firat.edu.tr/personel/yayinlar/fua_35/35_65804.pdf

5 < 1% match (Internet from 27-Jan-2017)
http://dergipark.ulakbim.gov.tr/etku/issue/download/5000018722/5000006551

6 < 1% match (publications)
Vennila Ramalingam, Susan Wiedenbeck. "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, 1999

7 < 1% match (student papers from 17-Jan-2019)
Submitted to Eastern Mediterranean University on 2019-01-17

8 < 1% match (Internet from 16-Nov-2018)
https://uknowledge.uky.edu/cgi/viewcontent.cgi?article=1001&context=accountancy_etds

9 < 1% match (Internet from 30-May-2016)
http://partedres.com/archieve/issue_1_1/Volume_1_Issue_1_Full.pdf

10 < 1% match (Internet from 08-Dec-2018)
http://www.temjournal.com/content/71/TemJournalFebruary2018_182_187.pdf

11 < 1% match (Internet from 22-Jun-2017)
http://research.wsulibs.wsu.edu:8080/xmlui/bitstream/handle/2376/12138/Carter_wsu_0251E_11707.pdf?isAllowed=y&sequence=1

12 < 1% match (Internet from 24-Jan-2015)
http://aut.researchgateway.ac.nz/bitstream/handle/10292/5667/WangLu.pdf?sequence=4