DEVELOPING MIDDLE SCHOOL STUDENTS' COMPUTATIONAL THINKING SKILLS USING UNPLUGGED COMPUTING ACTIVITIES

HAVVA DELAL

BOĞAZİÇİ UNIVERSITY

2019

DEVELOPING MIDDLE SCHOOL STUDENTS' COMPUTATIONAL

THINKING SKILLS USING UNPLUGGED COMPUTING ACTIVITIES

Thesis submitted to the

Institute for Graduate Studies in Social Sciences

in partial fulfillment of the requirements for the degree of

Master of Arts

in

Educational Technology

by

Havva Delal

Boğaziçi University

2019

DECLARATION OF ORIGINALITY

I, Havva Delal, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature...

Date ....18.07.2019.......................................

ABSTRACT

Developing Middle School Students' Computational Thinking Skills Using Unplugged Computing Activities

This study investigated the role of unplugged computing activities on developing computational thinking (CT) skills of 6th grade students. The unplugged computing classroom activities were based on Bebras challenges. Bebras, an international organization, aims to promote informatics and CT among school students. Participants of the study were 6th grade (n=24 female and n=29 male) students from two public middle schools in Istanbul. The activities in the study were divided into three groups as "easy", "medium" and "difficult" according to the difficulty levels as defined in the Bebras competition, which were selected so that each group involved four components of CT skills found to be common in CT definitions in the literature -- abstraction, decomposition, algorithmic thinking, and generalization. To evaluate students' CT skills, two equivalent tests were prepared. Questions in these tests were also selected from Bebras and translated into Turkish. Also, questions in the tests have three difficulty levels and covers the four major CT skills as in the activities. These tests were used as a pre-test and post-test, and their results were compared to assess students' CT skill development. The CT Scale (Korkmaz et al., 2016) was also given to the participants at the end of the instruction. The results of the study showed that students' post-test scores were significantly higher than pre-test scores. However, there was not any significant differences between students' scores in terms of gender, and there was no interaction effect between students' scores and their gender. In terms of the CT Scale results, there was no significant difference between male and female students' self-evaluation of CT skills at the end of the instruction.

ÖZET

Ortaokul Öğrencilerinin Bilgi İşlemsel Düşünme Becerilerinin Bilgisayarsız Bilgisayar Bilimi (B^3) Etkinlikleri ile Geliştirilmesi

Bu çalışmada, bilgisayarsız bilgisayar bilimi (B^3) etkinliklerinin 6.sınıf öğrencilerinin bilgi işlemsel düşünme becerilerinin geliştirilmesindeki rolü incelenmiştir. B^3 etkinlikleri Bebras yarışmalarından seçilmiştir. Bebras okul öğrencilerinin enformatik ve bilgi işlemsel düşünme becerilerini geliştirmek amacıyla düzenlenen uluslararası bir organizasyondur. Çalışmaya İstanbul'daki iki farklı okuldan 6. sınıf öğrencileri (24 kız ve 29 erkek) katılmıştır. Çalışmadaki sınıf içi etkinlikler Bebras'da olduğu gibi zorluk seviyelerine göre kolay, orta ve zor olmak üzere üç gruba bölünmüştür. Bu aktiviteler seçilmiştir, böylece her grup literatürdeki bilgi işlemsel düşünme tanımlarında sıklıkla kullanılan, soyutlama, ayrıştırma, algoritmik düşünme ve genelleme becerilerini içermiştir. Öğrencilerin bilgi işlemsel düşünme becerilerini ölçmek için iki paralel test hazırlanmıştır. Bu terslerdeki sorularda Bebras'dan seçilmiş ve Türkçeye çevrilmiştir. Aynı zamanda, testlerdeki sorular üç zorluk seviyesine sahiptir ve aktivitelerde olduğu gibi dört ana bilgi işlemsel düşünme becerisini kapsamaktadır. Bu testler ön test ve son test olarak kullanılmış ve sonuçları öğrencilerin bilgi işlemsel düşünme becerinin gelişimini değerlendirmek için karşılaştırılmıştır. Çalışmanın sonunda katılımcılara Bilgisayarca Düşünme Ölçeği de (Ortaokul Düzeyi İçin) (Korkmaz vd., 2016) verilmiştir. Araştırmanın sonuçları, öğrencilerin çalışma sonrası Bebras puanlarının çalışma öncesi Bebras puanlarından istatistiksel olarak anlamlı ve yüksek olduğunu göstermiştir. Ancak, öğrencilerin puanları arasında cinsiyet açısından anlamlı bir fark yoktur ve öğrencilerin puanları ile cinsiyetleri arasında herhangi bir etkileşim de

yoktur. Bilgisayarca Düşünme Ölçeği (Ortaokul Düzeyi İçin) sonuçları, erkek ve kız öğrencilerin, çalışma sonunda bilgi işlemsel düşünme becerilerini değerlendirme sonuçları arasında anlamlı bir fark olmadığını göstermiştir.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my advisor, Assoc. Prof. Diler Öner for her continuous support, patience, motivation, and encouragement. Without her expert guidance, support and productive feedback, the completion of this thesis would not be possible. I am really thankful for her to encourage me during this period.

I would like to offer my special thanks each of the other members of my thesis committee, Assist. Prof. Mutlu Şen Akbulut and Assoc. Prof. İbrahim Çetin, for their comments, feedbacks, and suggestions during this period. Especially I am grateful to Assist. Prof. Mutlu Şen Akbulut and Neslihan Er Amuce for reviewing the classroom activities.

Also, I thank my friends, Burak Buzkan and Tolga Açıkgöz, for helping me to reach the participants in their schools to conduct this research. In addition, my beloved friends Demet Turan, Hacer Yalçınkaya and Ayşe Arabacı made contributions to my research. Their productive comments and encouragement made this process easy for me. Specially, I would like to offer my greatest appreciation to Demet Turan for being there when I needed, providing language support and valuable feedback to this thesis.

Last but not least, I owe my sincere gratitude to my beloved family: to my mother, to my father, to my brothers and sister, and all of others. They always stood by me with their patience and encouragement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

CS: Computer Science

CSTA: Computer Science Teachers Association

CT: Computational Thinking

GPA: Grade Point Average

GUTS: Growing up Thinking Scientifically

ICT: Information and Communication Technology

ISTE: International Society for Technology in Education

ITEST: Innovative Technology Experiences for Students and Teachers

NAACE: The ICT association for advancing education through technology

NRC: National Research Council

RAPTOR: Rapid Algorithmic Prototyping Tool for Ordered Reasoning

CHAPTER 1

INTRODUCTION

Technology constitutes the majority of our lives in today's world. Computing is being used almost in every field and advances in computing makes our lives easy by providing solutions to urgent problems such as advances in preventing or curing the diseases, broadening our understanding about how our biological system works and facilitating our relationships to the world around us. However, although the technologies and tools used have changed from past to present, searching for solutions to problems always exist. Methods which are used to find solution to the problems included similar thinking skills throughout the history, even though the problems get more complex day by day. Some of these skills involve coping with uncertainty, reasoning, systematic thinking, continuing to try until reaching the result in challenging process, explaining the whole by analyzing the parts, and querying the problem by choosing the right variables (Kert, 2018).

Kert (2018) claimed that developing basic strategies is not required only for original discovery at the same time it can be required to solve lots of problems faced in daily life. To explain this, he gave an example; during a grocery shopping a customer who are looking for a cashier to pay makes a decision by looking at the queue in front of the cash register. At the same time s/he doesn't only look at the length of the line, s/he also looks at its pace. As a result, regardless of what kind of problem encountered, determining basic variables by using multifaceted sub-thinking skills and seeking to formulate the solution for repeated usage are common processes that also make up CT.

The term CT was firstly used by Saymour Papert (1996). While Papert was discussing how computers can be used in the solution of geometric problems, he claimed that CT can be used to explain the problem and its solution explaining the relationship between these two concepts. (Papert, 1996 cited as Çetin & Uçar, 2018). Although this term was first mentioned by Papert, it attracted the attention of society with the definition of Jeannette Wing (Wing, 2006). Wing (2006) defined CT as a skill that everyone could have, not just computer scientists. In her paper, she defined CT as “solving problems, designing systems and understanding human behavior by drawing on the concepts of computer science” (p.33). CT is relatively a new area and evolving rapidly. Thus, different researchers make their own CT definitions as a result of their research and there is no consensus on exact definition (Barr & Stephenson, 2011; Çetin & Uçar, 2018; Kalelioğlu, Gülbahar, & Kukul, 2016). However, there are some commonly used aspects used for the definitions of CT in the literature (Barr & Stephenson, 2011; Kalelioğlu et al., 2016; Çetin & Uçar, 2018).

As mentioned above, over the past few decades computer technology has taken place in everywhere. It is used in many different disciplines ranging from health to security, from art to sport and so on (Kert, 2018; Sayın 2018). With the development of technology, it is expected from everybody to have some basic computing skills. Being a digital citizen requires students to have CT skills as defined by ISTE (Kalelioğlu et al., 2016). In addition to this, Sayın (2018) stated that in order to be an educated citizen we need to understand CS and its basics, and it has been expected us to have problem solving skills which is a part of CT. Also, the same statements were included in the CSTA report (2011). To meet these needs, many countries around the world have begun to introduce CS training into their existing educational systems. With the CS education, it is not aimed to train computer scientists; it is aimed that everyone, especially children, will have the opportunity to produce new things and gain new skills by using computers. That is, there is a need for individuals who can use the power of computing to find solutions to the problems in every field (Sayın, 2018).

Although CS education was considered as an undergraduate level education by many, in recent years many countries give it a place in K-12 education programs (Bell, Andreae, & Lambert, 2010; Hubwieser, 2012; Brown et al., 2013). Turkey is one of these countries that gives priority to CS education starting at the middle school level. In the Turkish curriculum, computer education is provided under the name of "Bilişim Teknolojileri ve Yazılım", a compulsory course for 5th and 6th grades and offered as an elective for 7th and 8th grades (Tebliğler Dergisi, 2013).

In CSTA report, it was highlighted that today's students will continue to live a life, which is heavily influenced by computing, and many of them will work in the fields that directly involve computing. Therefore, they must begin to work with computational and problem-solving methods and tools in K-12 (CSTA, 2011). Also, in these systems CT took an important place because it was believed that CT was an area that has the possibility to contribute to personal and social development of people and to generate serious gains for national economies (Wing, 2008; Çetin & Uçar, 2018; Kert 2018).

Some researchers observed that many students experienced difficulties in CS because of having negative attitudes towards computer education (Bell, Alexander, Freeman, & Grimley, 2008). In this context, the results of the studies made with unplugged computing activities showed that these activities supported students' CT skills, changed their attitude towards CS in a positive way, improved their interests to CS, helped them to learn CS concepts and promoted a higher self-concept (Wohl, Pohler, & Clinch, 2015; Rodriguez, Rader, & Camp, 2016; Kalelioğlu, 2018). Unplugged computing activities are the activities that involve students physically and do not require to use computer to teach CS concepts (Nishida et al., 2009). Unplugged computing instruction and B^3 (in Turkish) are the terms used to refer to unplugged CS instruction.

Unplugged computing activities can also be used in order to increase students' creativity skills, make learning process more enjoyable and decrease the difficulties in the process (Nishida et al., 2009; Kalelioğlu, 2018). Of course, computer usage is required for programming and automation of solutions. However, to understand the importance of programming concepts and purpose students need to be prepared for it. Especially some CT concepts in CS such as problem solving, dividing problem into sub-problems, abstraction, explaining the solution in order, following the instructions, testing the solution and debugging it, while and if concepts are some of the CS topics that can be taught with unplugged computing activities (Kalelioğlu, 2018).

The gender role was not examined in detail in both computer-based and unplugged computing studies conducted to investigate the development of participants' CT skills (Carlisle et al., 2005; Lee, 2010; Brennan & Rescnick, 2012; Burke, 2012; Meerbaum-Salant et al., 2013; Thies & Vahrenhold, 2013; Lye & Koh, 2014; Wohl et al., 2015; Cortina, 2015). Therefore, there is a need to conduct more research investigating the gender role on the development CT skills.

According to Selby and Woollard (2013) CT can be defined as a mental tool set. Therefore, it is extremely important to teach this thinking skill, and there should be activities that all students actively take parts in. With unplugged computing activities several CT processes can successfully be acquired (Computing at School & NAACE, 2014; Curzon et al., 2014).

To sum up, as the importance of computing increasing, countries have started to give CS education a place in K-12 to have individuals being able to use the power of computing to find solutions to the problems in every field (Bell, Andreae, & Lambert, 2010; Hubwieser, 2012; Brown et al., 2013; Sayın, 2018). In CS education, CT has a crucial role since it can contribute to people's personal and social development (Wing, 2008; Çetin & Uçar, 2018; Kert 2018). In the literature, there was no consensus on the definition of CT; however, there were some terms which were frequently used in all CT definitions (Barr & Stephenson, 2011; Kalelioğlu et al., 2016; Çetin & Uçar, 2018). Furthermore, to develop the CT skills, unplugged computing activities can be used, which increase creativity, create an enjoyable learning environment and make learning process easier (Nishida et al., 2009; Kalelioğlu, 2018). However, there is lack of instructional materials in the Turkish curriculum for developing CT skills using unplugged computing activities and examining the role of gender on the development participants' CT skills, and there is a need for research on the role of gender and unplugged computing activities on students' CT skills development. Such instruction needs to be equally engaging for both boys and girls.

1.1 Purpose of the study

The purpose of the current study was to investigate the role of unplugged computing activities on developing students' CT skills by comparing the differences between students' pre-test scores and post-test scores. In addition, the study examined the differences between students' CT skills in terms of their gender and looked at the interaction between students' CT scores and their gender. Also, the study compared female and male students' test scores in terms of Korkmaz and colleagues' CT Scale (2016).

1.2 Research questions

This study was designed to answer the following research questions:

Research Question 1: Is there a significant difference between students' CT test scores before and after attending to the unplugged computing instruction?

Research Question 2: Is there a significant difference between male and female students' CT test scores?

Research Question 3: Does any interaction occur between participants' gender (male and female) and the time of pre- and post-tests?

Research Question 4: Is there a significant difference between male and female students' scores in terms of the CT Scale at the end of the treatment?

1.3 Research hypotheses

The research hypotheses for the first four questions are as follows:

Students will significantly develop their CT skills after attending unplugged computing instruction.

There will be no significant difference between male and female students' Bebras test scores due to their gender.

There will be no interaction between students' gender (male and female) and the time of the pre- and post- tests.

There will be no significant difference between male and female students' scores in terms of Korkmaz and colleagues' CT Scale.

CHAPTER 2

LITERATURE REVIEW

In the line with the purpose of examining the role of using CS unplugged computing activities on developing students' CT skills, the literature was reviewed. First of all, the information about the definition of CT was given and the sub-components of CT was explained. Then, the CT studies were reviewed including both the studies about computer-based and unplugged computing activities. The final part includes studies about how CT was assessed.

2.1 Definition of CT

The term CT was firstly used by Saymour Papert (1996); however, it has gotten more attention in recent years especially after being defined by Jeannette Wing in 2006. Many researchers have conducted studies about it in order to constitute a definition. However, there is little agreement on what CT entails, or how it should be taught and assessed (Barr & Stephenson, 2011; Kalelioğlu et al., 2016; Çetin & Uçar, 2018). In this part of the paper different CT definitions were presented.

Wing (2006) introduced CT and defined it as a skill that everyone could have, not just computer scientists. In her paper, she defined CT as "solving problems, designing systems and understanding human behavior by drawing on the concepts of computer science." (p. 33). Cuny, Synder, and Wing (2010) developed another definition for CT. They defined it as "the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent." (p. 1). According to Wing (2006), everyone can benefit from thinking computationally, and her grand vision is that CT will be a fundamental skill just like reading, writing, and arithmetic used by everyone by the middle of the 21st century. Moreover, the ISTE and CSTA (2011) produced an operational definition for CT. According to their definition, CT is a problem-solving process that has some characteristics: “formulating problems in a way that enables us to use a computer and other tools to help solve them, logically organizing and analyzing data, representing data through abstractions such as models and simulations, automating solutions through algorithmic thinking (a series of ordered steps), identifying, analyzing, and implementing possible solutions with the goal of achieving the most efficient and effective combination of steps and resources, generalizing and transferring this problem-solving process to a wide variety of problems” (p. 1).

Mannila and her colleagues (2014) stated that CT is a term encompassing a set of concepts and thinking processes that help in analyzing and organizing problems and their solutions in different disciplines in a way which could include computers. Similarly, Riley and Hunt (2014, p. 4) stated that “the best way to characterize computational thinking is as the way that computer scientists think, the manner in which they reason.”. Also, Sysło and Kwiatkowska (2013) addressed that CT covers a set of thinking skills that might not be included in computer programming. They claimed that CT should “focus on the principles of computing rather than on computer programming skills” (Syslo & Kwiatkowska, 2013 as cited in Kalelioğlu et al., 2016, p. 585).

Some researchers, such as Brennan and Resnick (2012) and Lye and Koh (2014), developed dimensions to define CT. The definition of Brennan and Resnick (2012) encompassed three dimensions which were computational concepts, computational practices, and computational perspectives (see Table 1).

Table 1. Brennan and Resnick's CT Dimensions

Dimension	Description	Example
Computational concepts	concepts that are used while coding	Variables: sequences, loops, parallelism, events, conditionals, operators, and data
Computational practices	practices that was developed while coding	being incremental and iterative, testing and debugging, reusing and remixing, and abstracting and modularizing
Computational perspectives	perspectives that users gained about the world and about themselves	Expressing, connecting, questioning

Source: [Brennan & Resnick, 2012]

Brennan and Resnick (2012) stated that computational concepts, which are generally common in other programming languages, are used by young people while programming interactive media and described seven CT concepts which are mostly used while programming in Scratch. These were sequences, loops, parallelism, events, conditionals, operators, and data. For computational practices, which was their second dimension, they claimed that only depending on computational thinking concepts was not enough to explain CT. They defined the design practices describing the processes of construction as the part of CT. Computational practices focus on how learners were learning rather than what they were learning. In computational practices, thinking and learning process were more important. They observed learners while they were programming interactive media and determined four sets of practices, which they were used mostly: being incremental and iterative, testing and debugging, reusing and remixing, and abstracting and modularizing. The last dimension, namely computational perspectives, was defined as designing interactive media with Scratch improves designers' understanding of themselves, their relationships to other and the technological world around them because while programming in Scratch students express their taught by designing or creating something and in Scratch students have possibility to reach new people and projects. Components of this perspective were expressing, connecting and questioning.

When we look at the definitions of CT in the literature, abstraction, problem, solving, algorithmic and thinking are used more frequently (Kalelioğlu et al., 2016). Kalelioğlu and her colleagues (2016) developed "Wordle" (see Figure 1), based on well-known definitions of CT made by researchers. This "Wordle" includes commonly used words in the definition of CT in the literature. Kalelioğlu and her colleagues (2016) stated that based on the Wordle, the most frequently mentioned words are abstraction, problem, solving, algorithmic and thinking.

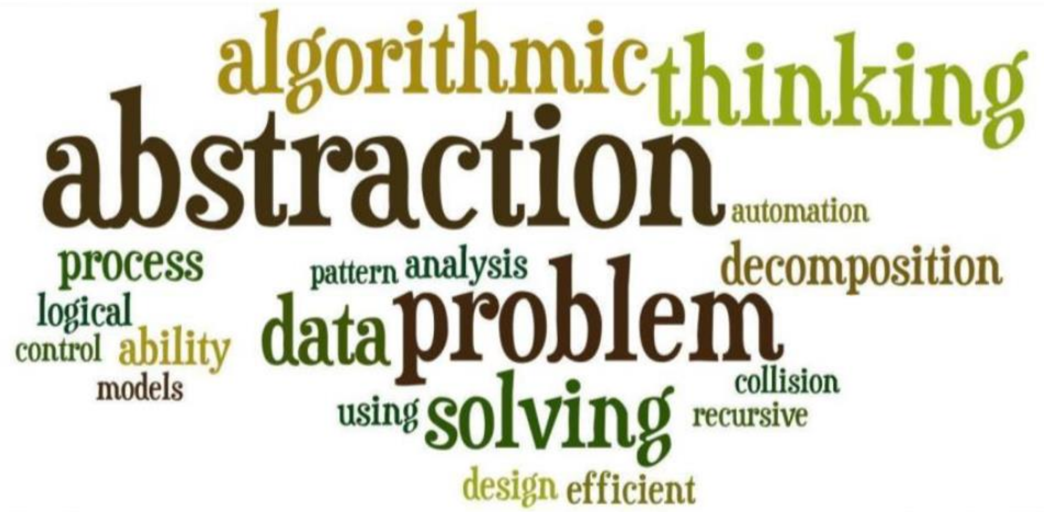


Figure 1. Wordle
Source: [Kalelioğlu et al., 2016]

In the current study, the CT definition of Selby and Woollard (2013) will be used since their definition covers commonly used terms in the definitions of CT in the literature. Also, categorization of the Bebras tasks are based on the definition of Selby and Woollard (Degiene et al., 2017). Selby and Woollard (2013) defined CT as "a cognitive or thought process that reflects: the ability to think in abstractions, the ability to think in terms of decomposition, the ability to think algorithmically, the ability to think in terms of evaluations, and the ability to think in generalizations" (p. 4).

They stated that this definition covers only the terms which were commonly used in literature and there is a consensus on the acceptance of them as being CT components, and they are well-defined across disciplines. Moreover, they claimed that "computational thinking is a focused approach to problem solving, incorporating thought processes that utilize abstraction, decomposition, algorithmic design, evaluation, and generalizations" (p. 5). These CT components were explained in detail below.

2.1.1 CT skills

2.1.1.1 Abstraction

Even though abstraction was introduced by Wing (2006) as being a part of CT, a clear definition has not been provided. Denning (2007) accepted that the essence of the computing field is abstraction. After one year later Wing (2008) supported this view by defining abstraction as the cornerstone of CT. Moreover, participants of the NRC agreed on that CT was centered around the process of creating and managing abstractions and defining the relationships between layers of abstraction (NRC, 2010). To express the levels of abstraction in the programming, Aharoni (2000) determined three thinking types which were programming language-oriented thinking, programming-oriented thinking and programming free thinking. In the first stage, students pay attention to the certain programing languages while searching solution to the problem. However, in the second stage, although they use the concepts of programing languages while solving the problem, they do not feel obliged to use a specific programming language. At last level, students can find a solution to the problem without needing any programing language and using any concepts of it.

Thanks to the abstraction skill, individuals can efficiently eliminate unnecessary details about the problem, realize key elements in the problem and choose a representation of the problem. It helps to simplify the problem by ignoring certain details of it (Dagiene, Sentence & Stupuriene, 2017; Liskov & Guttag, 2000).

2.1.1.2 Decomposition

Wing (2006, 2007) defined decomposition as a part of CT, which was defined as breaking a problem down into its components and each of them can be solved separately (Computing at School and NAACE, 2014). Decomposition is needed when dealing with large problems, complex systems, or complex tasks (Selby & Woollard, 2013) since taking the whole problem to solve mostly does not help the solution of it easily. Instead of taking the big problem, breaking it down into smaller components is preferred. Thus, the problem solving processed is simplified (Liskov & Guttag, 2000).

2.1.1.3 Algorithmic thinking

In the literature, CT is also linked to algorithmic thinking and there is a consensus on that CT includes aspects of algorithmic thinking (Selby & Woollard, 2013). The ITEST Working Group on CT (2011) picked out that CT "shares elements with various other types of thinking such as algorithmic thinking, engineering thinking, and mathematical thinking" (p. 32). Moreover, Cuny, Snyder and Wing (2010) stated that CT incorporates algorithmic thinking and parallel thinking. Algorithm was defined as following a procedure as step by step to accomplish a task and it was not just belonging to computer science; it was the same in all other disciplines (Selby & Woollard, 2013). Futschek (2006) defined algorithmic thinking as including different abilities to construct and understand an algorithm. These abilities were analyzing given problem, specifying a problem explicitly, finding the basic actions to given problem, forming a correct algorithm to a given problem using the basic actions, thinking about all possible cases of a problem and enhancing the efficiency of an algorithm. Furthermore, the members of the NRC committee defined algorithmic thinking as "…general concepts of algorithmic thinking, [that include] functional decomposition, repetition (iteration and/or recursion), basic data organizations (record, array, list), generalization and parameterization, algorithm vs. program, top-down design, and refinement" (Fluent 1999 cited as Cooper 2000, p. 11).

2.1.1.4 Generalization

Selby and Woollard (2013) identified generalization as the step of understanding how small parts can be reused and reapplied to similar or unique problems. It also took part in the definition of ISTE and CSTA (2011). One component of their definition included that generalizing and transferring the problem-solving process to a wide variety of problems.

According to Çetin and Uçar (2018), a new problem or new problems can be solved by constructing the patterns, finding the similarities and making connections in the solution of the specific problem. To do this we need to ask some questions to ourselves: How does the problem I solve earlier look like this new problem? Are there any connections between them? Can I use the strategies I have used to solve the previous problem for the solution of current problem? If it can be said that an algorithm or data structure used in the solution of a problem can be used to solve certain types of problems, generalization is realized.

2.1.1.5 Evaluation

According to Selby and Woollard (2013), analysis which is used in the context of a solution of a problem, is corresponding to evaluation and is used frequently in the literature. They included evaluation in their CT definition since they claimed that it was well defined across multiple disciplines. Evaluation was defined by Dagiene and his colleagues (2017) as finding the best solution to the problem, making decisions about using the best resources and being suit for purpose.

2.2 Developing CT

Computerized or unplugged computing activities were used to promote CT in the curriculum (Kalelioğlu et al., 2016). Although several studies reported the effects of computerized activities, less research is available on adapting unplugged computing activities for use in classroom (Kalelioğlu et. al., 2016; Rodriguez, Rader, & Camp, 2016). Some studies about CT and their results was presented below.

2.2.1 Computer-based studies to develop CT skills

Computer based projects were developed to improve CT skills of teenage and college students. A number of studies were conducted with middle, high and college students to assess the role of computer-based projects on the development of their CT skills. In these studies, the gender role was not examined in detail (Carlisle et al., 2005; Lee, 2010; Brennan & Rescnick, 2012; Burke, 2012; Meerbaum-Salant et al., 2013; Lye & Koh, 2014).

Burke (2012) stated that teaching computer programming to young children has been seen as difficult due to its abstract and complex nature. However, visual programing languages such as Scratch, Alice, RAPTOR and so on can make learning programing concepts easier for adolescents (Burke, 2012; Carlisle et al., 2005; Lee, 2010).

Lye and Koh (2014) argued that using visual programming languages (e.g., Alice) rather than traditional programming languages (e.g., Java or C++) was better to promote CT in K-12 contexts. This is because the commands are closer to spoken language and unnecessary syntax is eliminated such as using semicolon and curly brackets. In visual programming languages students only drag and drop blocks, and such programming tools help to reduce the cognitive load. Thus, students could focus on logic and structures in programming rather than thinking about the rules of how to write the code (Kelleher & Pausch, 2005). As such, these features of visual programming languages allow learners to learn the computational concepts more easily without having to learn complex programming syntax (Lye & Koh, 2014). Also, Cooper (2000) studied with Alice to support the development of algorithmic learning for beginner programmers; however, they did not draw any certain conclusion about it. They just stated that "Alice provides a natural set of problems to solve and an environment that supports teaching and developing algorithmic thinking in solving those problems" (Cooper, 2000, p. 5)

Another visual programing language is RAPTOR, which aids learners to develop algorithms. Carlisle and his colleagues (2005) claimed that teaching programming in RAPTOR improved problem-solving skills better than teaching programming in a more traditional, non-visual language.

Scratch is another programming language and have been used with different age groups to develop CT. Resnick and colleagues (2009) defined Scratch as an authoring environment which provides an opportunity for young people to design their own interactive media by gathering programming instruction blocks together. Thus, they might make stories, animations or simulations by using Scratch. Brennan and Rescnick (2012) and Meerbaum-Salant and his colleagues (2013) studied with Scratch to investigate whether it could be used to teach CS concepts. Their findings showed that Scratch was successful in developing CT skills and teaching CS concepts to students.

Apart from visual programing languages robotics for pre-college students and iGame and GUTS project for middle school students were used to develop students' CT skills (Lee et al., 2011; Kalelioğlu et al., 2016).

2.2.2 The use of unplugged computing activities to develop CT skills

Some researchers argued that CT can be developed with unplugged computing projects and those projects provide an easy and a fast way for students to get the ideas of CS without having to use computers (Bell et al., 2008; Bell et al., 2009; Cortina, 2015). The unplugged computing approach has become popular because of a declining number of students enrolling in CS although there is an increasing demand from employers for skilled CS students (Bell et al., 2009). Unplugged computing activities provide a chance for students to think more deeply about the concepts of

CS because they get away from the thought of computer, which is usually seen as a tool or toy (Bell et al., 2008; Bell et al., 2009). Another aim of unplugged computing approach is to decrease the gap between K-12 teachers who may not have technical background; yet, are supposed to teach technical ideas (Rodriguez, 2015).

Cortina (2015) stated that in unplugged computing activities, students take place in the activities physically and lots of activities encourage group work. By this way students work together to solve problems. At the same time, he stated that this approach promotes problem-solving and creativity. The activities are based on experiential learning, and students take part in the solution of the problem learning from observations and experiences. Contrary to some elementary programming activities which only give high importance to activity, unplugged computing activities bring students physically into the problem, and present an environment to work together, to share ideas and to design solutions (Cortina, 2015).

CT involves much more than learning how to program (Rodriguez, 2015); in addition to that, Lu and Fletcher (2009) asserted that CT can be isolated from programming and should be taught before programming instruction starts. According to Cortina (2015) unplugged computing activities support the principles of CT. When the concepts and applications of CT were examined, it is necessary to perform activities with block-based, text-based or physical programming tools especially for programming and automation of solutions. However, it may be needed to prepare students to better understand the importance and purpose of programming concepts. Especially problem solving, decomposition, abstraction, algorithmic thinking, following instructions, debugging and testing along with variables, loops and conditions can be taught without using computers (Kalelioğlu, 2018).

In the literature, there were some studies about the role of unplugged computing activities for the teaching of CS concepts. For instance, Curzon and his colleagues (2014) conducted a study about introducing teachers to CT by using unplugged computing activities and results showed that CT ideas can be successfully introduced by using unplugged computing activities. Furthermore, Thies and Vahrenhold (2013) stated that unplugged computing activities were as successful as conventional approaches in introducing the CS concepts and algorithms to lower secondary school students. Also, Wohl and his colleagues (2015) tried to teach CS concepts to kids aged between five to seven by using unplugged computing activities, Scrath and tangible computing. Their findings showed that students who studied with unplugged computing activities were more successful in understanding the concepts of algorithms and in logical predictions and debugging. In these studies, gender effect was not examined detailed.

Some unplugged computing projects stated in the literature are CSunplugged.org (Bell, et al., 2009), CS4FN (Curzon at al., 2009), Code.org unplugged, Informatic erLeben (Mittermeir, Bischof and Hodnigg, 2010 cited as Kalelioğlu, 2018), Abenteuer Informatic(Gallenbacher, 2012 cited as Kalelioğlu, 2018), Keşf@ (http://www.kesfetprojesi.org/kodlama) discovering coding and Bebras (https://www.bebras.org/) (Kalelioğlu, 2018). The common point of all these projects is to help students discover the concepts of CS without using the computer and to provide a positive attitude about CS (Kalelioğlu, 2018).

2.2.2.1 Bebras (Bilge Kunduz)

Bebras is an international challenge that was organized every year. Bebras community consists of members from different countries who are responsible for annually organizing a national Bebras challenge in their country. Many countries simultaneously participated in the online Bebras challenge to test students' informatics capabilities. Students do not need to have any obligatory prerequisite knowledge to attend this challenge (Dagiene & Stupuriene, 2016; www.bebras.org/?q=about). In Turkey, the Bebras challenge is called as "Bilge Kunduz" (http://www.bilgekunduz.org/yarisma-hakkinda/).

Bebras, acknowledged as in the categories of unplugged computing activities (Kalelioğlu, 2018), aims to motivate children to be interested in informatics (or computer science, or computing) (Dagiene & Futscheck, 2008). Especially the idea is to support the learning of students' informatics fundamental concepts and the development of algorithmic thinking as well as CT especially among school students of all ages (Dagiene, Stupuriene, Vinikiene, & Zakauskas, 2017). The challenge was organized for six aged groups: "Pre–Primary (grades 1–2), Little Beavers (grades 3–4), Benjamin (grades 5–6), Cadet (grades 7–8), Junior (grades 9–10), and Senior (grades 11–13)." (Izu et al., 2017, p. 42).

In the Bebras competition, short questions which can be answered in a few minutes are called Bebras tasks. The tasks were prepared by experts and delivered to Bebras community. Tasks can be created by researchers, teachers and students in each country. Proposed questions are evaluated in the annual Bebras task evaluation workshops. At least two reviewers from Bebras community were assigned for each offered task (Dagiene & Stupuriene, 2016; Dagiene et al., 2017).

Tasks in the challenge are applied on the computer. They are multiple-choice and interactive. Multiple choice tasks consist of four well-defined answer choices with only one correct solution. A scene or a diagram is presented on the computer screen for interactive tasks that participants can interact with to carry out actions and change the scene's state to move towards the question's goal. Also, the tasks in the challenge have different levels of difficulty as A (easy), B (medium) and C (hard) (Dagiene, Stupuriene, & Vinikiene, 2017; Izu et al., 2017). The prepared tasks are collected in a pool, and each country annually chooses tasks from this pool. Usually, every country has its own national committees to choose and organize this challenge for their countries (https://www.bebras.org/?q=workshops).

2.2.2.1.1 Bebras task categorization system

A categorization system for Bebras tasks was not indicated before 2016. First categorization system was proposed in 2006. It included seven categories which were "general logic, ICT in everyday life, practical and technical issues, information comprehension, algorithms and programming, mathematics underlying computer science, and history and trivia." (Opmanis et al., 2006, p. 3).

These categories were also used in developing new tasks; however, few years later the Bebras tasks categorization were revised and in the new categorization system, there were six categories which were "Information comprehension, Algorithmic thinking, Structures, patterns and arrangements, Puzzles (logical), Using computer system, Social, ethical, cultural, international and legal issues." (Dagiene & Futscheck, 2008, p. 21).

This categorization model has been used to the present day; however, there were limitations of this categorization system. For example, researchers point out that it was difficult to distribute Bebras tasks across categories since many tasks might belong to several categories, and some categories overlapped (Dagiene, Sentence, & Stupuriene, 2017). In 2009, Kalas and Tomcsanyiova proposed an alternative categorization for Bebras tasks. It consisted of only four categories, which were "digital literacy, programming, problem solving, data handling." (Kalas & Tomcsanyiova, 2009, p. 3)

This categorization system was seen as too general and also involving overlapping categories. Thus, it has not been adapted in categorizing tasks. In 2017, Dagiene and her colleagues (2017) presented a new categorization system by taking into consideration the fact that completing a Bebras task demonstrates skills in CT. In this new two-dimensional approach tasks were classified as knowledge level, which includes informatic concepts, and skills level that includes CT. In the knowledge level, tasks were classified according to their content. They determined five categories for content which school informatics should include. These were "Algorithms and programming, including logical reasoning, Data, data structures and representation (includes graphs, data mining), Computer process and hardware, Communications and network, Interaction systems and society." (Dagiene et al., 2017, p. 35-36). The other dimension was CT, skills level. Their presented categorization of CT skills is based on the work of Selby and Woollard (2013). The CT definition of Selby and Woollard (2013) consisted of five CT components, "abstraction, algorithmic thinking, decomposition, evaluation and generalization." (Dagiene et al., 2017, p. 37). In the current study, skills level categorization was considered while selecting both the classroom tasks and assessment items from Bebras challenges.

Working group for UK Bebras challenge publish answer key sheets for each task, where they write which CT skills the task covers (Howarth, Millican, Roffey and Sentance, 2014).

To sum up, different classification models were proposed for Bebras tasks throughout the years (Opmanis et al., 2006; Dagiene & Futscheck, 2008; Kalas & Tomcsanyiova, 2009; Dagiene et al., 2017). Currently, Dagiene and her colleagues' two-dimensional classification model is being used (Dagiene et al., 2017).

2.2.2.1.2 Role of gender in Bebras challenges

In recent years, participants of Bebras challenge has been markedly growing (Dagiene et al., 2017). As mentioned before, this event is designed to promote pupils' interest about informatics and aimed to be equally engaging for both boys and girls. Although informatics is still male-dominated, Dagiene and colleagues (2017) claimed that in the worldwide more than %40 of participants of Bebras event are girls. Furthermore, girls aged between 10 and 13 were even more successful than boys in this event (Dagiene et al., 2014). However, some study results showed that there were no significant differences in the interests and performance of boys and girls (Kalaš and Tomcsányiová, 2009).

Hubwieser and Mühling (2015) investigated 2009 Bebras challenge in Germany. They reported that boys performed significantly better compared to girls. In addition, another research on the performance of participants in the German Bebras challenge of 2014 demonstrated that boys were more successful than girls and the difference between their performances increased dramatically with the age of the participants. However, it was detected that in some tasks (related to real life situation and easy to solve) female participants performed better (Hubwieser et al., 2016). On the other hand, according to Izu and colleagues (2016) in some tasks, boys performed better than girls. Those tasks required spatial abilities and depended on participants having an intuitive understanding.

In Turkey, Korkmaz and colleagues (2015) examined participants' CT skills in terms of different variables (school type, department, age, gender and so on). They used the CT Scale developed by Korkmaz and colleagues (2015) as an instrument for their study. They studied university students and found that only in critical thinking questions male students got higher points. In other words, in critical thinking male students ranked themselves more sufficient. In all other sub-categories, there was not a difference between male and female students. Also, Bilge Kunduz 2015 report among 5th and 6th grade students showed that female students were more successful than male students in 2015 Bebras event in Turkey (Gülbahar, Kalelioğlu, & Doğan, 2016). However, when we examined Bilge Kunduz 2017 report, it can be seen that, boys performed better than girls as age-level increases (see Figure 2).

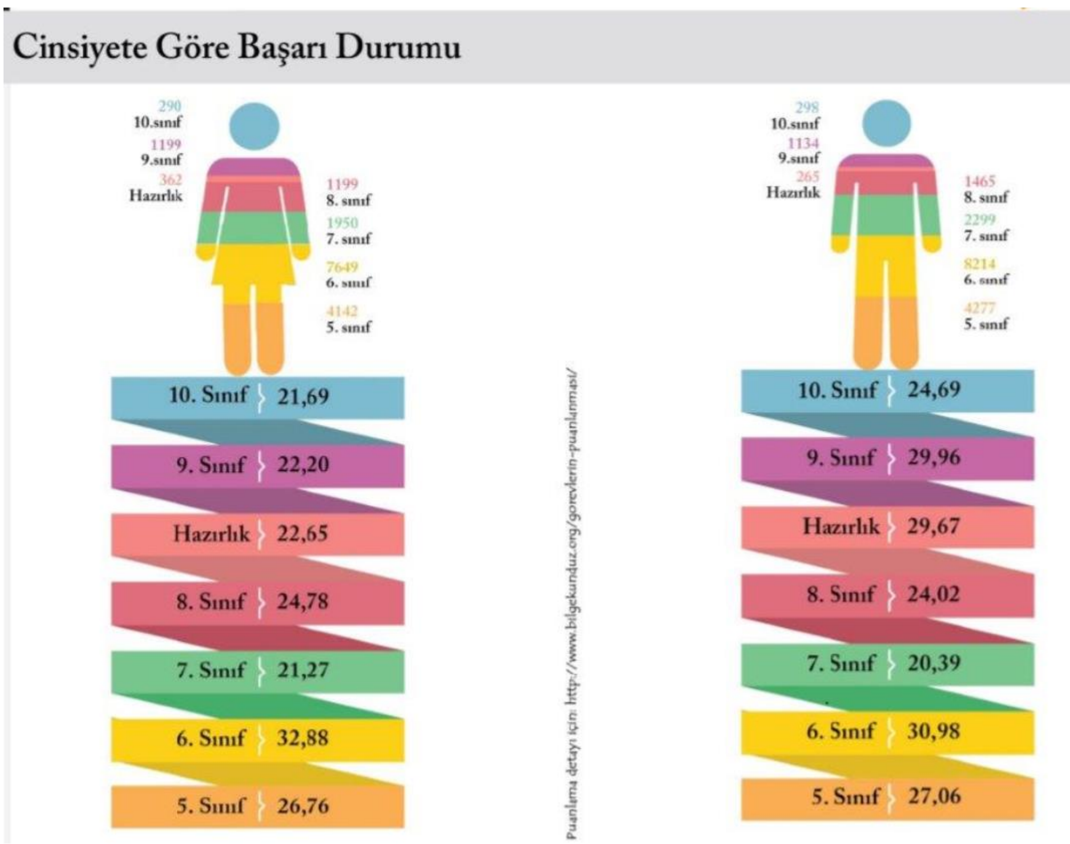


Figure 2. Success results based on gender

Source: ["Bilge Kunduz 2017 report", 2017]

2.3 Strategies to evaluate CT skills

In recent years, the assessment of CT skills is one of the most discussed topics. This section reviews research focusing to measure the development of CT.

2.3.1 Assessment of computerized projects

To assess the development of CT in young people who are programming interactive media with Scratch, Brennan and Resnick (2012) developed three approaches, project portfolio analysis, artifact-base interviews and design scenarios. In portfolio analysis, Breanna and Resnik (2012) described seven CT concepts which are mostly used while programming in Scratch. These were sequences, loops, parallelism, events, conditionals, operators, and data. They used Scrape tool (the "User Analysis" tool - http://happyanalyzing.com/) to assess the CT, which was developed to analyze the programming blocks used in Scratch projects. With the help of the "User Analysis" tool they could identify the computational concepts that were used within Scratch projects. This approach focuses just on the blocks used in the projects. It does not give information about the processes of developing CT practices. The second approach was artifact-based interviews. They claimed that only depending on CT concepts is not enough to explain the learning and participation of learners. They defined the design practices describing the processes of construction as the part of CT. They observed learners while they were programming interactive media and determined four sets of practices: being incremental and iterative, testing and debugging, reusing and remixing, and abstracting and modularizing. They interviewed learners and created a discussion environment for learners to discuss how the developing process of the projects. Therefore, they claimed that this approach gives opportunities to assess how designers are applying CT practices.

Their final approach was using design scenarios. For this approach they developed three sets of Scratch projects which had increasing complexity. Each set included two different projects, consisting of the same concepts and practices. Students choose one projects from each set. Then, students are asked to explain what the selected project does. Secondly, they are asked to offer ideas about how it could be extended. Thirdly they are asked to fix a bug if there is any and lastly, they remix the project by adding a feature. In this approach, students' critique, extend, debug and remix the projects. Thus, they use different concepts and practices.

Another strategy was suggested by Wilson, Hainey and Connolly (2012) to assess the development of the programming concepts that could be learned while designing interactive media with Scratch. They constituted three main categories and 22 subcategories (see Table 2). Main categories involved programming concepts, organization of code, and design of code for usability. Projects are coded for the presence of each sub element (either 0/1) or in some cases the extent to which that element was used within the categories using a range from either 0-2 or 0-3.

Also, Seiter and Foreman (2013) developed a model called PECT (Progression of Early CT) by extending and using the coding scheme developed by Wilson, Hainey and Connoly (2012). Moreover, Moreno-León and Robles (2015) developed a plug-in for the Hairball program, called Mastery, which assessed the CT skills in Scratch projects automatically. They specified seven concepts for the assessment of CT, which are abstraction and problem decomposition, parallelism, logical thinking, synchronization, algorithmic notions of flow control, user interactivity and data representation. To evaluate each of these concepts, they developed an algorithm for Mastery depending on some rules (see Table 3). The Mastery plug-in generates a CT score depending on the developers' competence on these seven defined concepts. They evaluated each of CT concepts in three levels: basic, developing and proficiency, and each level had a score. 7 points represented basic. 8 and 14 points were marked as developing and more than 15 points were evaluated as proficient. The total CT score was calculated by summing up these partial scores.

Table 2. Wilson, Hainey and Connolly's Assessment Rubric

Programming concepts found in Scratch		Coding
Sequence	Are the blocks in a systematic order to execute the program correctly?	0/1
Iteration	Using forever and repeat to create iterations.	0-3
Variables	Variables can be created within Scratch and then be used within programs.	0-3
Conditional Statements	Using if, forever if and if-else to check for conditions.	0-3
Lists (arrays)	Allows for storing and accessing lists of strings and numbers.	0/1
Event handling	Responding to events triggered by either the user or another script.	0-2
Threads	Launching two independent scripts at the same time to execute in parallel.	0-2
Coordination and Synchronisation	Using blocks such as wait, broadcast and when I receive to coordinate the actions of multiple sprites.	0-3
Keyboard Input	Using blocks such as ask and wait prompts users to type in an answer.	0-2
Random Numbers	Pick Random is used to select random integers within any given range.	0/1
Boolean Logic	Using and, or, not.	0/1
Dynamic Interaction	Using mouse x or y and loudness can be used as dynamic input for interaction.	0/1
User Interface Design	Using when sprite clicked button can create an interactive user interface.	0/1
Code organisation		
Extraneous blocks	Are there any blocks which are not initialised when the program is run?	-1/0
Sprite names (the default is overridden).	Are the default sprite names overridden?	0/1
Variable names	Are the variables given meaningful names when set up?	0/1
Designing for usability		
Functionality	Does the game run when it is started (most games start when the green flag is clicked)?	0-3
Goal	Is there a clear defined goal to the game?	0-2
Sprite customisation	Is the sprite used a predefined sprite or has the sprite been customised and to what extent.	0-3
Stage customisation	Is the stage used a predefined stage or has the stage been customised and to what extent.	0-3
Instructions clear	Has the student defined how the game is supposed to run?	0-3
Game originality	Students were asked to create a maze game to give them the grounding in basic skills that were required. However when it came to creating their own game students were able to adapt the maze game or create a new game entirely.	0-3

Source: [Wilson, Hainey, & Connolly, 2012]

Furthermore, they have developed a web tool called Dr. Scratch. Users could upload their projects which had sb2 file extension or give the URL of the project. Dr.

Scratch tool analyzed the project and generated an overall CT score. Dr. Scratch informs users about the partial scores which they took from different CT concepts (see Figure 3).

The researchers indicated some limitations about their tools. One of them was examining only one project might not enough to obtain a reliable result about CT skills. Moreover, some key CT competences such as the debugging or remixing skills could not be measured. Also, looking at whether block was used or not does not show they were used correctly. Therefore, in the future version of it they were planning to use some plug-in, Dead code, Attribute initialization, Sprite naming or Repeated code, to check whether block was used correctly or not. Also, users could create an account and save their analyzed records. Thus, they could follow their development (Moreno-León & Robles, 2015).

Table 3. Level of Development of CT

CT Concept	Basic	Developing	Proficiency
Abstraction and problem decomposition	More than one script and more than one sprite	Definition of blocks	Use of clones
Parallelism	Two scripts on green flag	Two scripts on key pressed, two scripts on sprite clicked on the same sprite	Two scripts on when I receive message, create clone, two scripts when %s is > %s, two scripts on when backdrop change to
Logical thinking	If	If else	Logic operations
Synchronization	Wait	Broadcast, when I receive message, stop all, stop program, stop programs sprite	Wait until, when backdrop change to, broadcast and wait
Flow control	Sequence of blocks	Repeat, forever	Repeat until
User Interactivity	Green flag	Key pressed, sprite clicked, ask and wait, mouse blocks	When %s is >%s, video, audio
Data representation	Modifiers of sprites properties	Operations on variables	Operations on lists

Source: [Moreno-León & Robles, 2015]

Figure 3. Dr.Scratch shows the CT score after analyzing a Scratch project

Source: [Moreno-León & Robles, 2015]

2.3.2 Assessment of unplugged computing projects

There is less research available about unplugged computing activities in the literature and they usually did not consist any method to evaluate unplugged computing projects (Rodriguez, Rader, & Camp, 2016). Researchers mostly worked with unplugged worksheet activities. To assess them they created a rubric. The rubric provided guidelines on how to score worksheet answers as either proficient, partially proficient, or unsatisfactory (Curzon & McOwan, 2008; Taub et al., 2009; Wohl et al., 2015; Rodriguez, Rader, & Camp, 2016).

Curzon and McOwan (2008) conducted a study to teach basic CS concepts to students aged 11-17. They created unplugged computing activities that included links between various tricks and CS. For assessment, at the end they used questionnaire sheets to take students' feedback, and they stated that they received lots of positive feedbacks. Taub and colleagues (2009) studied with unplugged computing activities (developed by Bell et al., 1998). To examine the role of activities they used a questionnaire including 22 Likert-type statements to understand students' views and attitudes and an interview.

Wohl and his colleagues (2015) tried to teach CS concepts by using unplugged computing activities, Scrath and tangible computing. They used paper models and interviews to assess students' logical predictions about algorithms. At the end of the study they found that students who studied with unplugged computing activities were more successful in understanding the concepts of algorithms and in logical predictions and debugging.

In summary, the assessment of unplugged computing projects mostly included assessment of attitudes using questionnaires.

2.4 Summary

Various CT definitions were proposed by researchers after the first CT definition offered by Wing in 2006. While Wing and her colleagues (2010) defined CT as the thought processes that included the formulation of a problem and its solution (thus, the solution of the problem can be easily performed by human beings or a tool) some researchers developed different dimensions to define CT (Brennan & Resnick, 2012; Lye & Koh, 2014). It was stated that CT was not just related to the way computer scientists think. It was also seen as a skill that everyone can have (Wing, 2006).

Although even today there is still less agreement on the exact definition of it, some commonly used components in the CT definitions were determined (Selby and Woollard, 2013; Kalelioğlu et al., 2016). Selby and Woollard (2013) identified the most frequently used CT components as, abstraction, decomposition, algorithmic thinking, generalization and evaluation.

Both computer-based and unplugged computing activities can be used to develop CT. For computer-based studies, robotics, iGames and visual programming languages such as Scratch and Alice were used. Researchers reported that visual programming languages and other tools (robotics and iGames) were successful in teaching CS concepts to students and developing CT skills (Cooper, 2000; Carlisle et al., 2005; Lee, 2010; Lee et al., 2011; Brennan & Rescnick, 2012; Burke, 2012; Meerbaum-Salant et al., 2013; Lye & Koh, 2014; Kalelioğlu et al., 2016). To evaluate computer-based projects some approaches were used which mostly included a rubric and a tool (e.g. Dr.Scratch, Mastery) that determined the programming concepts used in the developed projects (Brennan and Resnick, 2012; Wilson, Hainey, & Connolly, 2012; Seiter & Foreman, 2013; Moreno-León & Robles, 2015).

Some researchers claimed that CT comprises more than coding (Rodriguez, 2015) and should be taught before programming (Lu & Fletcher, 2009). Some accepted unplugged computing actives more appropriate for teaching CT (Kalelioğlu, 2018). They claimed that unplugged computing activities provide an environment in which students physically took part in and worked together. Also, unplugged computing activities made the way easy to understand the key points of CS and the aim of programming concepts better (Bell et al., 2008; Bell et al., 2009; Lu & Fletcher, 2009; Cortina, 2015; Rodriguez, 2015; Kalelioğlu, 2018). Some unplugged computing activities mentioned in the literature were CSunplugged.org, CS4FN, Bebras and so on (Bell, et al., 2009; Curzon at al., 2009; Kalelioğlu, 2018). The studies made with unplugged computing activities indicated that they were successful in increasing students' interest about CS and in teaching basic CS concepts; however, it was stated that there is less research about using unplugged computing activities in classrooms (Rodriguez, Rader, & Camp, 2016).

Bebras is an international challenge that is carried out every year by the member countries. It intended to reach different age groups to advocate informatics and CT among them. It includes tasks consisting of different difficulty levels (easy, medium and difficult). Throughout the years different categorization systems were developed to classify the tasks in the Bebras. Recently, two-dimensional classification method is in use. One dimension of the categorization system is knowledge level. The other one is CT skills. CT skills dimension depends on Selby and Woollard's (2013) CT definition. It includes five CT components, abstraction, decomposition, algorithmic thinking, generalization and evaluation. It was expected that while students are solving a single Bebras task, they will develop these defined CT skills.

In addition to these, it was claimed that Bebras tasks were not gender biased. It aims to be equally engaging both for girls and boys (Dagiene et al, 2017; Izu et al, 2017). However, in some studies, it was presented that boys were more successful than girls (Hubwieser & Mühling, 2015). In others, girls were more successful than boys (Dagiene et al., 2014) Also, in some, it was stated that there was no difference between boys' and girls' performances (Kalaš and Tomcsányiová, 2009). These discrepancies were tried to be explained by some researchers. They claimed that it can depend on the task content. That is, some tasks can be more interesting for boys or girls (Izu et al., 2016). For all of these reasons, there is a need to conduct more research investigating whether there was a difference between boys' and girls' CT scores depending on the tasks.

As the importance of computing evolving, countries have begun to give importance to CT in their curriculum (Bell, Andreae, & Lambert, 2010; Hubwieser, 2012; Brown et al., 2013; Sayın, 2018). However, there is no study about developing students' CT skills by using unplugged computing activities in Turkish curriculum. The current study aims to close this gap in the literature.

In this study, Bebras tasks were used as unplugged computing classroom activities. Ten activities were compiled from Bebras challenges encompassing three difficulty levels covering commonly used CT components. Also, for each activity an explanation sheet was prepared for classroom use. In addition to these, two equivalent tests were prepared to assess the development of students' CT skills. These tests were used as pre- and post- test in the study. Question in these tests were also selected from Bebras challenges. They involved three difficulty levels enclosing the same CT components for each difficulty group as in the activities. Also, Korkmaz and colleagues' (2016) CT Scale was used, which was developed by the researchers for self-evaluation of students' their CT skills. In this study four research questions were examined: Is there a significant difference between students' Bebras test scores before and after attending to the unplugged computing instruction? Is there a significant difference between male and female students' Bebras test scores? Does any interaction occur between participants' gender (male and female) and the time of pre- and post-tests? Is there a significant difference between male and female students' scores in terms of the CT Scale at the end of the treatment?

Depending on the presented literature, it is expected that students will significantly improve their CT skills after participating the unplugged computing classroom instruction. There will be no significant difference between male and female students' Bebras test scores due to their gender and there will be no interaction between students' gender (male and female) and the time of the pre- and post- tests. Also, there will be no significant difference between male and female students' scores in terms of Korkmaz and colleagues' CT Scale.

CHAPTER 3

METHOD

In this chapter, research design, sampling and participants of the study, details about the treatment carried out, data collection instruments, and data collection and analysis procedures were explained.

3.1 Research design

In this study, one group pre-test post-test design was used (Creswell, 2003). In this design, first a pre-test is given to a group, after the pre-test, they take a treatment, and then a post-test is given (Sekaran, 1992). According to Tharenou and colleagues (2007) even though this design is commonly used, researchers must be careful when interpreting the effects of the treatment.

The independent variable of the study was the unplugged computing instruction that involved tasks selected from Bebras challenges to develop middle school students' CT skills. The dependent variable of the study is CT skills of students as measured by two tests whose items were also compiled from Bebras challenges. The difference between post-test scores and pre-test scores shows their CT improvement scores. Students were also given the CT Scale (Korkmaz et al., 2016) at the end of instruction.

3.2 Sampling and participants

The target population of the study was 6th grade public middle school students in Turkey. 6th grade ICT curriculum includes problem-solving, developing an algorithm to solve a problem, and introduction to programing topics (Milli Eğitim Bakanlığı, 2018). In the current study, 6^{th} grade public school students in Istanbul were determined as the accessible population.

The participants (n = 53) were chosen from two public middle schools in Istanbul located in Bağcılar (school 1) and Küçükçekmece (school 2). Purposive sampling, a non-probability sampling method, was used. In this sampling method, researcher determines what needs to be known and finds participants who can and are willing to provide information for the study (Bernard, 2002). Those two schools have similar student profile in terms of socio-economic status and students' success levels. To determine their success levels, the previous semester general GPAs were examined using the Turkey Ministry of National Education's e-school platform (82.5 and 81.90 for school 1 and school 2 respectively). Also, the two groups' pre-test CT scores (measured by the pre-test developed in this study) were compared using independent t-test. The results showed that there were no significant differences between the scores of the students from the two schools, $t(51) = -.52$, $p = > .05$. That is, their CT skills level can be considered similar before the instruction. According to these results, it was assumed that these two groups of students were similar on the variables associated with the treatment. Twenty-four 6^{th} grade female students from the first school and twenty-nine 6^{th} grade male students from a second school participated in this study. Principals of the schools and students' parents were informed about the aim of the research, and students were free to participate in the research or not.

3.3 Treatment: Unplugged computing instruction

In the literature, there is a lack of unplugged computing activities and materials in Turkish that can be used in the curriculum to develop students' CT skills. This study with the prepared unplugged computing activities can fulfill this gap in the ICT curriculum in Turkey. Ten activities were selected from the Bebras international challenge. The items were revised and translated into Turkish by taking two experts' opinions. The Bebras tasks were selected by taking into account four CT components (abstraction, decomposition, algorithmic thinking and generalization) which were commonly used in the literature (Barendsen et al., 2015; ISTE & CSTA, 2011; Kalelioğlu et al., 2016; Selby & Woollard, 2013). The four CT skills are used in this study based on Selby and Wollard's (2013) CT definition, abstraction, decomposition, algorithmic thinking and generalization. Since we were not able to find enough number of tasks in every difficulty category corresponding to the skill of evaluation, we did not address the evaluation skill in the instruction. Specifically, the CT definition of Selby and Woollard (2013) is taken into consideration because Dagiene, Sentence and Strupuriene (2017) stated that Bebras tasks were categorized depending on Selby and Woollard's CT definition, and this definition includes frequently used CT components.

The class activities were determined so that they had three difficulty levels, easy, medium and hard as labelled in the Bebras competition. Four of the chosen tasks were easy, three of them were in medium difficulty category, and three of them were hard. Moreover, the activities in each category approximately involved the same distribution of CT components, abstraction, decomposition, algorithmic thinking and generalization (see Table 4).

Table 4. Information About the Unplugged Computing Activities/Tasks

Implementation Order of the Activities	Name of the Activity	Difficulty Level	CT Component Covered	Bebras Challenge Year
1	Animation - Sıralama	Easy	Abstraction, Decomposition, Generalization	2015
2	Magical Bracelet – Bileklik	Easy	Decomposition Generalization	2014
3	Erase Walls – Duvar Kırma	Easy	Abstraction, Algorithmic Thinking	2017
4	Party Guests – Parti Daveti	Easy	Algorithmic Thinking, Decomposition	2016
5	Beavers on the Run – Ormanda Yürüyüş	Medium	Abstraction Generalization Algorithmic thinking	2014
6	Traffic in the City – Yol Haritası	Medium	Abstraction, Algorithmic Thinking	2014
7	Footprints – Baskı	Medium	Abstraction, Algorithmic Thinking, Decomposition	2014
8	Puddle Jumping – Zıplama	Hard	Abstraction, Algorithmic Thinking, Decomposition	2014
9	Meeting Point – Buluşma Yeri	Hard	Abstraction, Algorithmic Thinking, Decomposition, Generalization	2014
10	Social Network – Tanışma	Hard	Algorithmic Thinking, Generalization	2014

For each activity an explanation sheet was prepared for instructors. Each sheet included explanations about the activity such as difficulty level of it, estimated time, which CT component it meets and the role of students and teacher in this activity (see Appendix A). During the instruction, the activities were presented from easy to difficult.

During the implementation, the instructor (the researcher) introduced the activity and what it included, and then gave students some time to work on it. The researcher directed students, created a collaborative learning environment and helped them if it was necessary. In some activities (1, 8 and 9 listed in the Table 4), the researcher asked them to work as teams. At the end of each activity students discussed what they found and how they found it. Then, one from each group were asked to explain their solution to the whole class. Finally, the researcher summarized the main points of the activity to the whole class.

An example activity from the instruction is given below.

3.3.1 Beavers on the run (activity 5)

"Beavers on the run" activity was selected from the Bebras 2014 challenge and it was in the easy category. The activity consists of a story that involves 5 beavers who were labeled as A, B, C, D, and E who went to a forest for a walk and there were 4 different depth pits on their way. 4 beavers fit into in the first pit, 2 beavers fit into in the second pit, and 3 beavers fit into in the third pit. Blue lines on the Figure 4 showed how many beavers could fit into the pit. Rules for the crossing pits of beavers were given and the question asked the order of beavers was at the end of the road. (Question mark on the Figure 4 showed final position place).

Before starting this activity, the instructor is asked to go over a mini activity about beavers who walked into the forest (see Figure 5) as written in the activity explanation sheet (see Appendix A), and students discuss their solutions. After the discussion, the sheet of "Beavers on the Run" is given to students (see Figure 5) and the instructor announces the time for this activity. Then students start to solve the activity by themselves. After they finished, they discuss their solutions and one is asked to explain his/her solution to the whole class.

Ormanda Yürüyüş

A, B, C, D, E olarak adlandırılan 5 kunduz, ormana yürümeye gitmiş. İlerledikleri yolda 4 tane farklı derinlikte çukur var. İlk çukura 4 kunduz, 2. çukura 2 kunduz, 3. çukura 3 kunduz sığabiliyor. Şekil 2 üzerinde verilen mavi çizgiler çukura kaç kunduz inebileceğini gösterir. Kunduzların çukurları geçme kuralları verilmiş. Bu kurala göre yolun sonunda kunduzların sırası nasıl olur? (Şekil 2'deki soru işareti kunduzların son konumlarının olacağı yeri gösterir.)

Kurallar:

1. Kaç tane kunduz çukura inebilirse iner.
2. Geri kalan kunduzlar çukurdan karşıya geçer.
3. Çukurdaki kunduzlarda çıkıp yola devam eder.

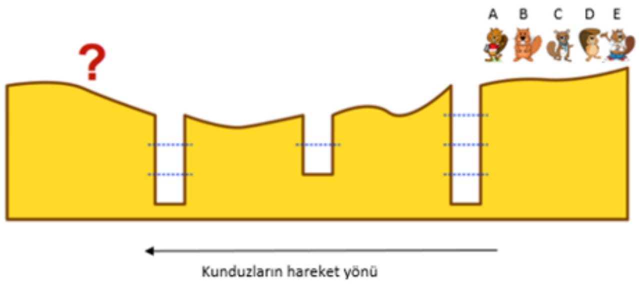


Figure 4. Walking in the forest activity sheet

With this activity, it is expected that students use abstraction, decomposition and algorithmic thinking skills (the CT components guiding this study). In CS, organizing data in a structured way is important. There are many data structures to organize the data and stack is one of them. This task is an example of a stack, which works similar to stacking plates on top of each other. New plates are always added on top of the stack and have to be removed from the top one at a time. This type of structure is generally called as a LIFO structure – the objects that have been added last are the first to be removed. (LIFO = Last In First Out) (Howarth at al., 2014).

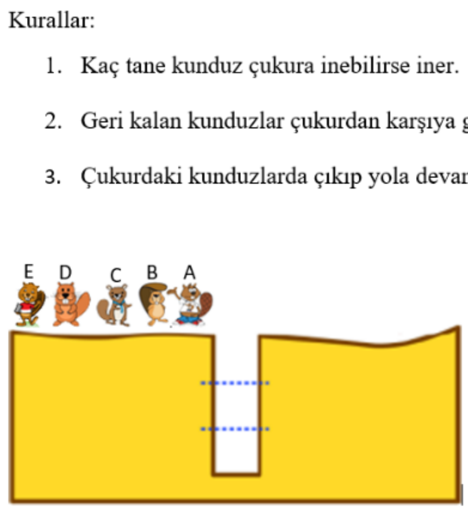


Figure 5. A mini walking forest activity

3.4 Data collection instruments

Using the CT skill categories and difficulty levels, two parallel tests were prepared to assess the development of CT skills of participants. Questions in the tests were selected from Bebras challenges and translated into Turkish by taking three experts' opinion. These parallel tests included questions from three different difficulty levels (5 easy, 5 medium, and 5 difficult), and each question in each group were assigned to one or more CT skills being defined by Selby and Woollard (2013). One of these tests was used as the pre-test (see Appendix B) and the other one was used as the post-test (see Appendix C). Also, the CT Scale which was developed by Korkmaz and colleagues (2016) was given to the students at the end of the instruction (see Appendix D).

3.4.1 The pre-test

As mentioned above, 15 questions were selected from Bebras challenges, which included different difficulty levels and a set of CT skills (abstraction, decomposition, algorithmic thinking and generalization). 5 of them were in easy category, 5 of them were in medium difficulty category and 5 of them in difficult category. The questions selected for each difficulty level category included the same distribution of CT skills as in the activities being used in the instruction. Information about the pre-test was given in Table 5. It includes the question name, difficulty level, which CT component it included and which Bebras challenge year it was taken.

Table 5. The Pre-test

Question Number	Question Name	Difficulty Level	CT Component Covered	Bebras Challenge Year
1	Telefon Tuşları	Easy	Algorithmic thinking, Decomposition	2014
2	Mesaj İletimi	Easy	Abstraction, Generalization	2017
3	Otopark	Medium	Algorithmic thinking, Decomposition	2017
4	Kılıç Kalkan Oyunu	Medium	Algorithmic thinking, Decomposition	2017
5	Hayvanlar yarışıyor	Easy	Abstraction Decomposition, Generalization	2015
6	Buz Hokeyi	Hard	Algorithmic thinking	2016
7	Şeker Labirenti	Medium	Abstraction, Generalization	2017
8	Bilyeler	Hard	Abstraction, Algorithmic thinking	2017
9	Sofra Düzeni	Easy	Algorithmic thinking	2015
10	Dizi Mesafesi	Medium	Algorithmic thinking, Generalization	2017
11	Okul Gazetesi	Medium	Abstraction, Algorithmic thinking	2017
12	Şifre Çarkı	Hard	Algorithmic thinking, Generalization	2017
13	Mantar Toplama	Easy	Abstraction, Algorithmic thinking	2015
14	Görüntü İşleme	Hard	Decomposition, Generalization	2017
15	Paketleme Makinesi	Hard	Abstraction, Algorithmic thinking, Decomposition	2015

No reliability and validity information of the Bebras tests were reported in the literature. Experts from different countries (these can be researchers, teachers and students) had prepared these questions and delivered to Bebras (detail information was given in the literature section). Prepared questions were evaluated in the annual Bebras task evaluation workshops (Dagiene & Stupuriene, 2016). Therefore, although there were no reliability and validity tests reported, there is an evaluation process conducted by Bebras. For the current study, the tasks were translated into Turkish by taking three different experts' opinion. One of these experts was a faculty member at Boğaziçi University who studies CT development, one of them was working as an educational technologist, and one of them was a software developer.

3.4.2 The post-test

Questions equivalent to the pre-test were selected for the post-test. As in the pre-test it included 15 questions selected from Bebras challenges using the same parameters. 5 of them were in easy category, 5 of them were in medium difficulty category and 5 of them in difficult category. The questions selected for each difficulty level category included the same distribution of the CT skills as in the activities being used in the instruction. Information about the post-test was given in Table 6. It includes the question name, difficulty level, which CT component it included and which Bebras challenge year it was taken.

3.4.3 Korkmaz and colleagues' CT scale

The CT Scale was first developed by Korkmaz, Çakır and Özden (2015) for university students. Then, they adapted it for secondary school students. It was called as “Bilgisayarca Düşünme Ölçeği (Ortaokul Düzeyi İçin)” in Turkish (see Appendix D). Korkmaz and colleagues (2016) claimed that it can be used to determine students’ self-evaluation of CT skills.

Table 6. The Post-test

Question Number	Question Name	Difficulty Level	CT Component Covered	Bebras Challenge Year
1	Dondurma	Easy	Abstraction, Generalization	2014
2	Baraj	Easy	Abstraction	2015
3	Süsleme Kağıdı	Medium	Abstraction, Generalization	2016
4	Çubuk Fındık Oyunu	Medium	Abstraction, Algorithmic thinking, Generalization	2015
5	Baskı	Easy	Algorithmic thinking, Decomposition, Generalization	2016
6	Gizemli Kareler	Medium	Abstraction, Algorithmic thinking	2016
7	Araba Yolculuğu	Medium	Algorithmic thinking, Decomposition	2016
8	Üçgen Şekiller	Hard	Algorithmic thinking, Generalization	2016
9	Kunduz Turnuvası	Easy	Abstraction, Algorithmic thinking	2017
10	Sihirli Kaplar	Hard	Algorithmic thinking	2016
11	Özel Tarif	Medium	Algorithmic thinking, Decomposition	2016
12	Hızlı Kunduz Kodu	Hard	Abstraction, Decomposition	2015
13	Beş Çubuk	Easy	Abstraction, Algorithmic thinking, Decomposition	2017
14	Hazine Arama	Hard	Abstraction, Decomposition	2015
15	Posta Kodu	Hard	Algorithmic thinking, Generalization	2016

CT Scale for Middle School is a five-point Likert type scale and consists of 22 items that can be grouped under five factors which are creativity, algorithmic thinking, cooperativity, critical thinking and problem solving. Response choices of the scale are always/ generally/ sometimes/ rarely/ never (Korkmaz et al., 2016).The reliability and the validity of the scale for secondary school has been examined and results showed that it is a valid test. The scale has good reliability since Cronbach's Alpha values of it is 0.809 (Korkmaz, Çakır, & Özden, 2016).

In the current study, this scale was used to check whether any significant difference occurred between female and male students' scores at the end of the treatment.

3.5 Data collection procedures

The approvals were taken from the sub-Ethics Committee of the Boğaziçi University (see Appendix E) and the administration of the participant schools before conducting the research study. Teachers and students were informed about the study. Consent forms were distributed to the students before the intervention.

The researcher carried out the instruction at both schools during their ICT classes. Before students worked with tasks, the pre-test was given to students. It lasted one class hour, approximately 40 minutes. After one week later, the treatment started and, in both groups, it took three class hours (approximately 120 min.), 2 class hours were in the same day, and one class hour was one week later. In the same week of one instruction, the post-test and Korkmaz and colleagues' CT Scale were given respectively. These tests took two class hours.

Throughout the instruction, the researcher explained the tasks, facilitated student work, directed class discussions and summarized what the major points were when it was necessary. At the implementation in both classrooms, it was aimed that students take an active role in all activities. Some tasks included group work, and in those cases the researcher grouped students. After all tasks are finished, the post-test was administered to assess students' development of CT skills. Then, Korkmaz and colleagues' CT Scale was given.

3.6 Data Analysis

Quantitative analyses were carried out in order to answer the research questions. To analyze the data which was gathered from the pre-test, the post-test and the CT Scale developed by Korkmaz and colleagues (2016) (Turkish: Bilgisayarca Düşünme Ölçeği), the IBM SPSS statistical software (Statistical Package for Social Sciences - Version 20) was used.

First of all, each student's pre- and post-test Bebras scores (1point for correct answer, 0 point for wrong answer), and Korkmaz and colleagues' CT Scale scores (from 1- strongly disagree to 5- strongly agree) were scored appropriately. Some items (1, 2, 3, 4, 5, 6 in the problem-solving group) in Korkmaz and colleagues' CT Scale included negative arguments. Therefore, the scores of those items reversed as 1→5, 2→4, 3→3, 4→2, 5→1. Next, each student's scores on each item were added and total scores were calculated.

In this study, two groups' (male and female) pre-test and post-test scores were measured at two time points before and after the unplugged computing instruction. Since the study aimed to investigate two main effects (pre/post-test as time and gender) with two levels and the interaction between them, 2 x 2 mixed design ANOVA was used to answer the first three questions.

Research Question 1: Is there a significant difference between students' Bebras test scores before and after attending to the unplugged computing instruction? – time main effect

Research Question 2: Is there a significant difference between male and female students' Bebras test scores? – gender main effect

Research Question 3: Does any interaction occur between participants' gender (male and female) and the time of pre- and post-tests? - interaction between time and gender

Between-subject factor included two groups (male and female). Within-subject factor included the same group of students since their Bebras scores were measured before and after attending the unplugged computing instruction.

Table 7. The Variables of the Two-Way Mixed Design ANOVA

	Female	Male
Pre Test (Time 1)		
Post Test (Time 2)		

The horizontal cells indicate that the same participants have taken equivalent forms of two Bebras tests.

To answer the 4th question "Is there a significant difference between male and female students' scores in terms of the CT Scale at the end of the treatment?" descriptive statistics including mean and standard deviation and a t-test were used to determine the differences between female and male students' responses to the CT scale.

CHAPTER 4

RESULTS

This part consists of the results of the analyses conducted to answer the four research questions. Both descriptive and inferential statistics were reported. In this study, to answer the first three research questions a 2 x 2 mixed design ANOVA (one between subject and one within subject) was used. Between-subject factors were gender (female and male) and within-subject factor was the time (including two time points) of the Bebras scores (pre-test and post-test).

Before carrying out the ANOVA test, parametric test's assumptions were controlled, which are normal distribution, homogeneity of variance, random sampling, independence of observations, and level of measurement (Pallant, 2007). Level of measurement assumption was verified since students' CT development was measured with Bebras scores (with compiled Bebras tasks). Furthermore, one measurement did not impact by the other one, thus independence of observation assumption was also satisfied. However, the random sampling was not assumed because the purposive sampling was adopted for this study.

In order to check the normality of the dependent variable, skewness and kurtosis values were calculated. The acceptable range for skewness and kurtosis is between -1.96 and 1.96. In Table 8, the skewness and kurtosis values of students' scores are shown. According to the results, students' skewness and kurtosis values for pre and post-test scores were in the acceptable range except the post scores of male students. The kurtosis value was 2.31 (SE = .84).

Table 8. Descriptive Statistics for Students' Pre-Post Test Scores

		N	Skewness	Kurtosis	Min	Max	Mean	Median	SD
Pre Test	Female	24	-.11	-.97	0	9	4.83	5	2.64
	Male	29	-.35	.98	0	10	5.17	5	2.13
Post Test	Female	24	.31	.62	2	10	5.38	5	1.97
	Male	29	.65	2.31	2	13	6.41	6	2.13

In addition, Shapiro-Wilk test was found to be not significant, which also showed that the data can be assumed to be normally distributed. The p values were all bigger than .05 (Table 9).

Table 9. Shapiro-Wilk Test

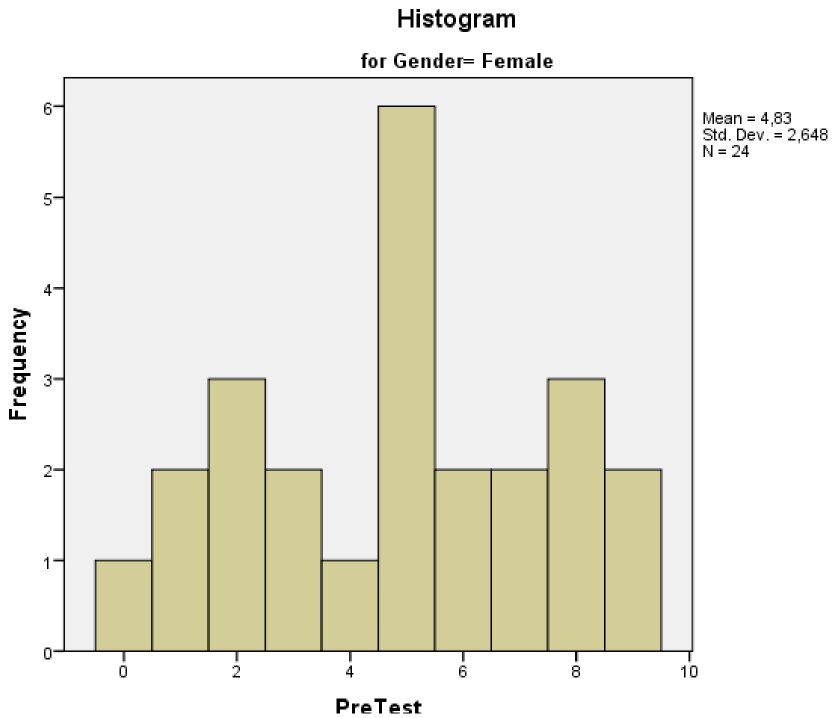
		Statistic	df	Sig.
Pre-test	Female	.95	24	.29
	Male	.93	29	.07
Post- test	Female	.93	24	.11
	Male	.93	29	.09

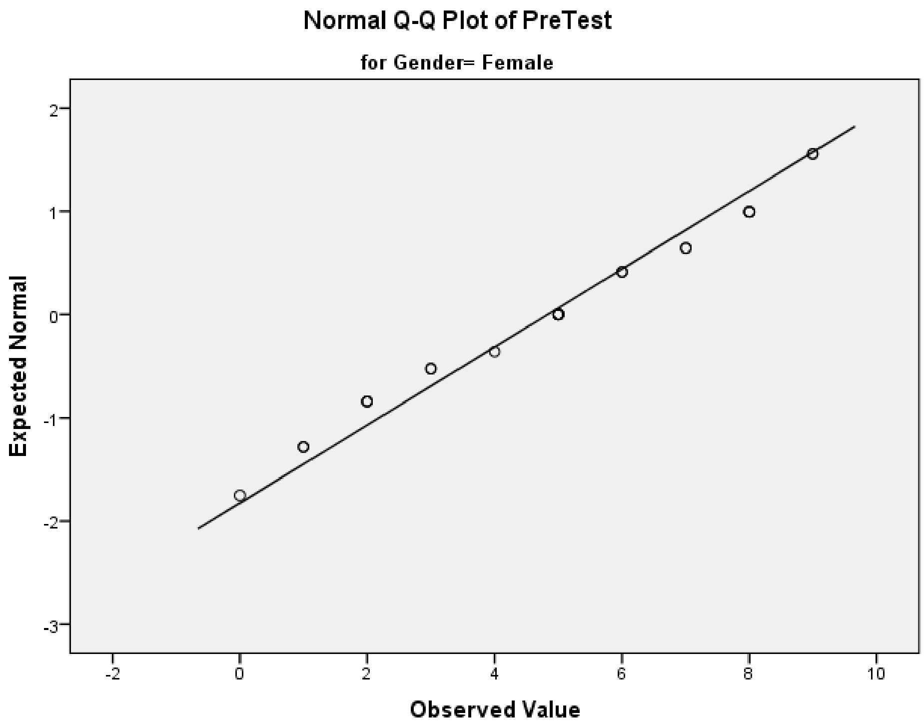
The Levene's test showed that the assumption of homogeneity of variance was met for pre- test scores, $p = .139 > .05$ and post-test scores, $p = .748 > .05$ (Table 10).

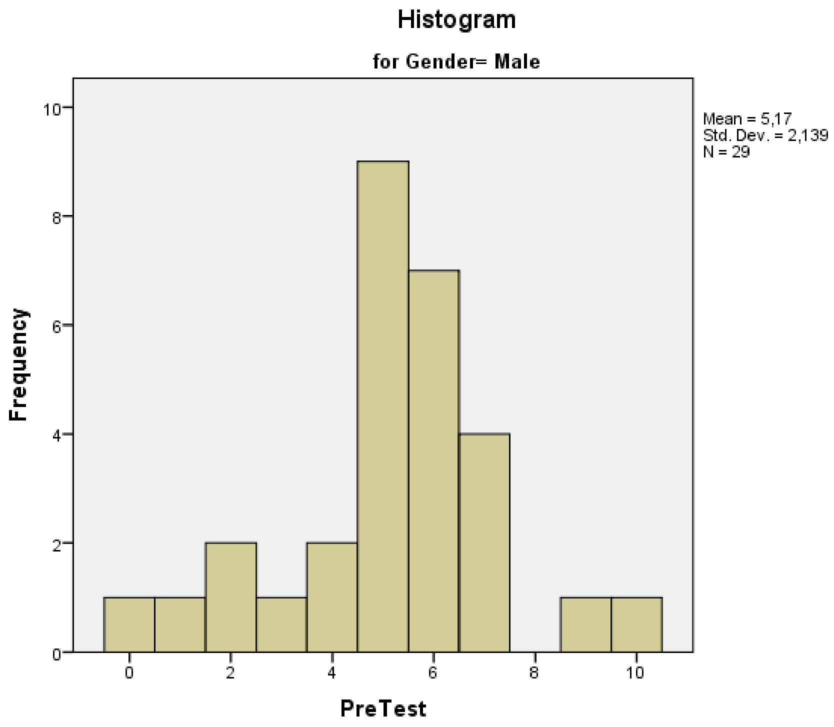
Table 10. Levene's Test for Equality of Variances

	Levene Statistic	df1	df2	Sig.
Pre-test	2.255	1	51	.139
Post-test	.104	1	51	.748

Moreover, the histograms and Q-Q plots given below support the normality of pre- and post-test scores distributions (Figures 6 and 7).


Histogram
for Gender= Female
Mean = 4,83
Std. Dev. = 2,648
N = 24
Frequency
PreTest
0
1
2
3
4
5
6
8
10


Normal Q-Q Plot of PreTest
for Gender= Female
Expected Normal
Observed Value
2
1
0
-1
-2
-3
4
6
8
10



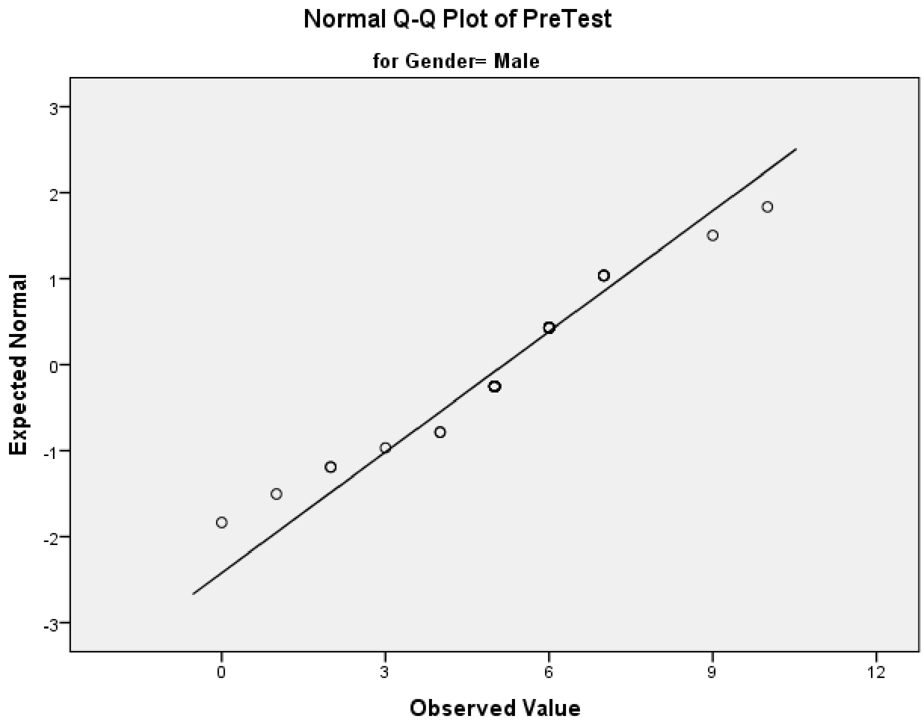
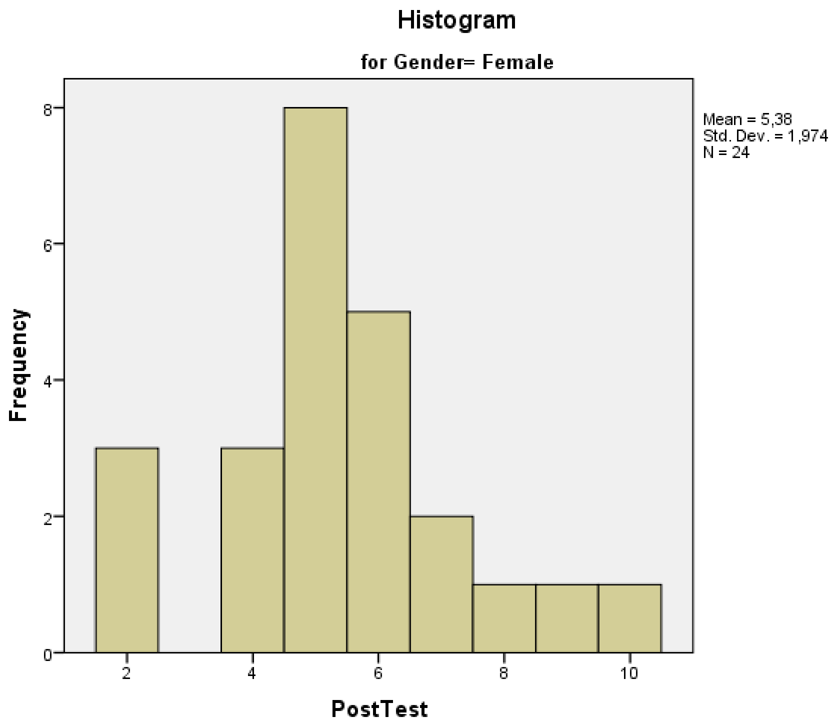
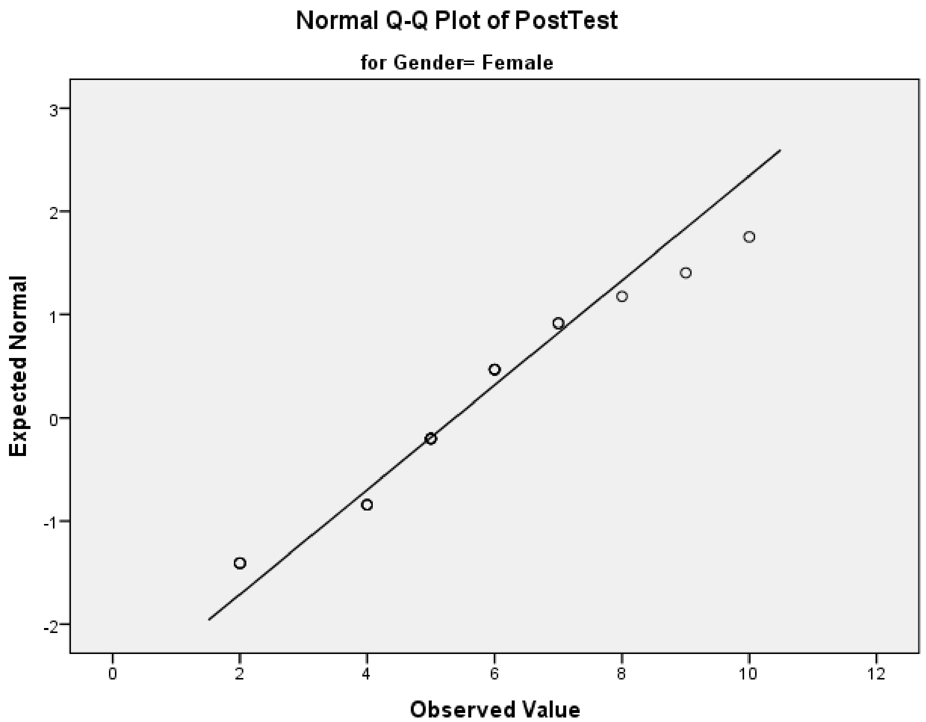


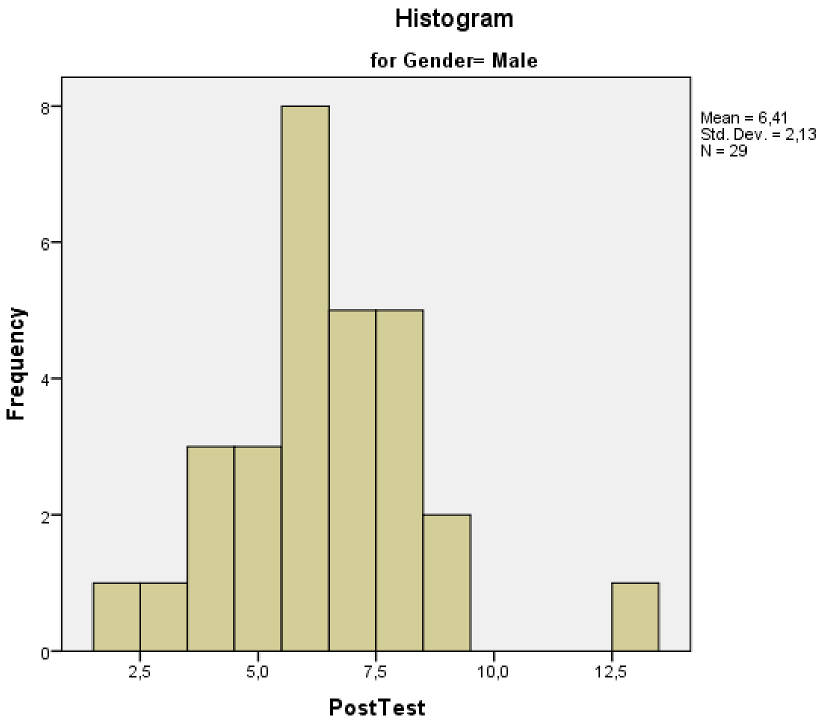
Figure 6. Histogram and Q-Q plot of pre-test Bebras scores



Histogram
for Gender= Female
Mean = 5,38
Std. Dev. = 1,974
N = 24
Frequency
PostTest
0
2
4
6
8
10



Normal Q-Q Plot of PostTest
for Gender= Female
Expected Normal
Observed Value
3
2
1
0
-1
-2
0
2
4
6
8
10
12



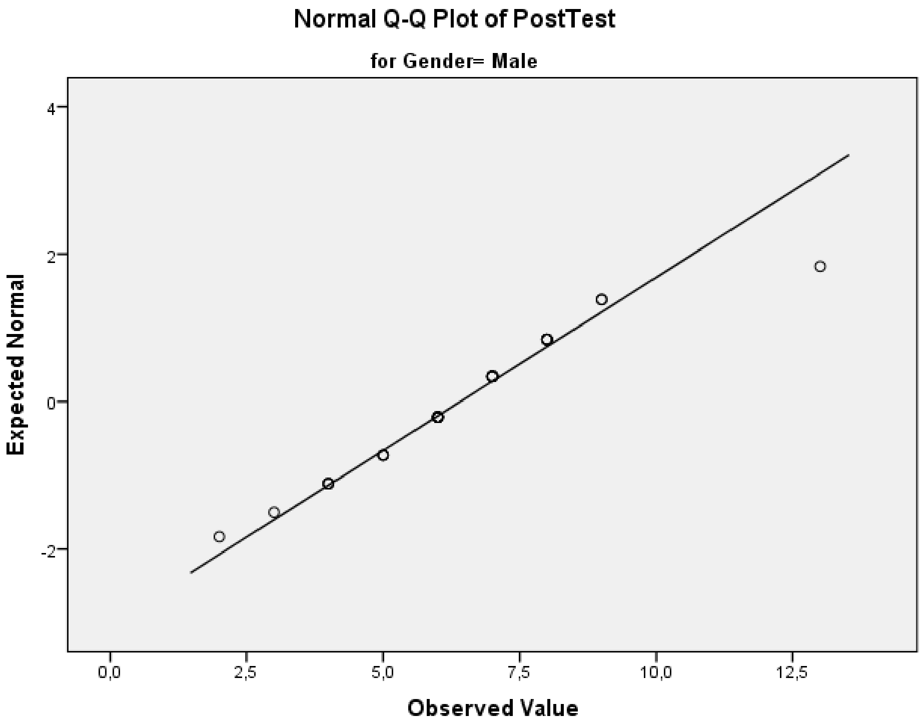


Figure 7. Histogram and Q-Q plot of post-test Bebras scores

4.1 Change in students' Bebras test scores before and after participating unplugged computing instruction

Research Question 1: Is there a significant difference between students' Bebras test scores before and after attending to the unplugged computing instruction?

As mentioned before, the unplugged computing instruction lasted three class hours, and fifty-three students attended to it. Over the three class hours, participants' CT success level was measured at two different time points (at the beginning of the instruction (pre-test) and at the end of the instruction (post-test). Table 11 shows that the means Bebras scores increased from 5.02 to 5.94.

Table 11. Descriptive Statistics for Within Subject Factor (Time)

	Mean	Std. Deviation	N
Pre-test	5.02	2.366	53
Post-test	5.94	2.107	53

A 2x2 mixed group factorial ANOVA analysis was conducted to investigate the significance of the change in students' Bebras scores as explained in the data analysis section. The first research question is concerned with the within subject analysis because the data were gathered from the same participants at two different points.

Results showed that there was a significant difference between participants' pre and post test scores, $F = 6.67$, $df = 1.00$, $p < .05$ (see Table 12). In other words, participants' Bebras scores significantly increased after attending the unplugged computing instruction. Cohen's effect size value ($d = .41$) suggested an effect between small effect (d =.2) and medium effect (d =.5) (Cohen, 1988).

Table 12. Test of Within-Subject Effects

Source		Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Time	Sphericity Assumed	20.875	1	20.875	6.669	. 013	.116
	Greenhouse-Geisser	20,875	1	20.875	6.669	. 013	.116
	Huynh-Feldt	20.875	1	20.875	6.669	. 013	.116
	Lower-bound	20.875	1	20.875	6.669	. 013	.116
Error (Time)	Sphericity Assumed	159.634	51	3.130			
	Greenhouse-Geisser	159.634	51	3.130			
	Huynh-Feldt	159.634	51	3.130			
	Lower-bound	159.634	51	3.130			

4.2 Does gender matter?

Research Question 2: Is there a significant difference between male and female students' Bebras test scores?

As explained before, participants were divided into two groups as female and male according to their gender, which represents the between-subjects factor (see Table 13). The between-subject effect was checked in order to see whether there was a significant difference between female and male participants' Bebras test scores (see Table 14).

Table 13. Descriptive Statistics of Between Subject Factors

Gender	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
Female	5.10	.37	4.34	5.85
Male	5.79	.34	5.10	6.48

Table 14. Test of Between-Subject Statistics

Source	Type III Sum of Squares	df	Mean Square	F	Sig.
Intercept	3118.881	1	3118.881	459.061	.00
Participants' Gender	12.466	1	12.466	1.835	.182
Error	346.496	51	6.794		

The descriptive statistics results showed that male mean scores (5.79) was higher than the female mean scores (5.10); however, inferential statistics indicated there was no significant difference between students' Bebras test scores regarding their gender (female and male), $F = 1.83$, $df = 1$, $p >. 05$ (see Table 14).

As we did not see any significant differences between female and male students' scores, we further looked at whether male and female students' scores differed within each difficulty level descriptively.

4.2.1 Change in male and female students' scores according to questions' difficulty level

As mentioned before, each of the test (pre and post) included fifteen questions, and five of them were in easy group, five of them were in medium difficulty group and the other five of them were in difficult group. This part examined students' Bebras scores according to questions' difficulty levels to answer the question whether there is a difference between female and male students' Bebras scores in terms of questions' difficulty levels. Students' mean scores for each difficulty levels were shown in Table 15.

Table 15. Students' Mean Scores for Each Difficulty Level of Bebras Tests

		Mean Scores for Each Difficulty Levels		
		Easy	Medium	Difficult
Pre-test	Female	2.50	1.58	0.62
	Male	2.72	1.68	0.75
Post-test	Female	2.66	1.54	1.16
	Male	3.55	1.96	0.89

4.2.1.1 Comparing scores in easy difficulty level questions

Question 1: Is there a difference between female and male students' Bebras scores in easy group questions?

As shown in the Table 15, while female students' pre-test mean score in easy questions was 2.5, male students' pre-test score in easy questions was 2.72. After the treatment, the means of both groups increased. While female students' post-test mean was 2.66, male students' post-test mean was 3.55. Male group's development is slightly higher (difference between means = 0.83) compared to the female group (difference between means = 0.16). Also, gained scores of students from easy questions were shown in Table 16.

Table 16. Gained Scores From Easy Questions

Participants	Post-test – Pre-test
Female	0.16
Male	0.83

4.2.1.2 Comparing scores in medium difficulty level questions

Question 2: Is there a difference between female and male students' Bebras scores in medium difficulty level questions?

Table 15 showed that as difficulty level of questions increased, the level of students' success decreased. In medium difficulty group questions, while the mean score of female students in the pre-test was 1.58, the mean score of male students in the pre-test was 1.68. In the post-test, although male students increased their mean scores (*M* = 1.96), female students slightly decreased their mean scores (*M* = 1.54). The increase of male students' mean scores was 0.18 points, and the decrease of female students' mean scores was 0.04. Also, in Table 17, students gained scores from the middle question was given.

Table 17. Gained Scores From Medium Questions

Participants	Post-test - Pre-test
Female	- 0.04
Male	0.18

4.2.1.3 Comparing scores in difficult level questions

Question 3: Is there a difference between female and male students' Bebras scores in difficulty level questions?

While, in the pre-test, female students' mean score was .62 and male students' mean score was .75, in the post test, female students' mean score was 1.16 and male students' mean score was .89. Female group's improvement is higher (difference between means = 0.54) compared to the male group (difference between means = 0.14). Thus, female students showed more improvement in hard group questions compared to the male students (see Table 18).

Table 18. Gained Scores From Difficult Questions

Participants	Post-test - Pre-test
Female	0.54
Male	0.14

4.3 Differential effect of unplugged computing activities on gender

Research Question 3: Does any interaction occur between participants' gender (male and female) and the time of pre- and post-tests?

Descriptive statistics indicated that female students' mean of pre-test scores was 4.83 and mean of post-test scores was 5.83. Male students' mean of pre-test scores was 5.17 and mean of post-test scores was 6.41 (see Table 19). To analyze whether students' pre and post test scores changed regarding their gender, the interaction effect between the "group" and the "time" of the Bebras test was examined (Table 20).

Even though there was a significant main effect of time $F = 6.669$, $df = 1.00$, $p < .05$ (see Table 12), there was no significant interaction between time and participants' gender, $F = 1.027$, $df = 1.00$, $p > .05$ (see Table 20 and Figure 8). In other words, the improvement in the level of CT skills can be considered homogenous for both groups of students. While female students started with lower level of CT skills ($M = 4.83$) and male students with higher CT skills ($M = 5.17$), both groups improved their CT skills after the instruction. Male group's improvement is slightly higher (gained score = 36) compared to the female group (gained score = 13) (Figure 8).

Table 19. Descriptive Statistics of Between and Within Subject Factors

Participants' Gender	Time	Mean	Std. Error	95% Confidence Interval Lower Bound	Upper Bound
Female	pre	4.833	.486	3.857	5.810
	post	5.375	.421	4.530	6.220
Male	pre	5.172	.442	4.284	6.061
	post	6.414	.383	5.645	7.182

Table 20. Test of Interaction Effects

Source		Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Time* Gender	Sphericity Assumed	3.215	1	3.215	1.027	.316	.020
	Greenhouse-Geisser	3.215	1	3.215	1.027	.316	.020
	Huynh-Feldt	3.215	1	3.215	1.027	.316	.020
	Lower-bound	3.215	1	3.215	1.027	.316	.020
Error (Time)	Sphericity Assumed	159.634	51	3.130			
	Greenhouse-Geisser	159.634	51	3.130			
	Huynh-Feldt	159.634	51	3.130			
	Lower-bound	159.634	51	3.130			

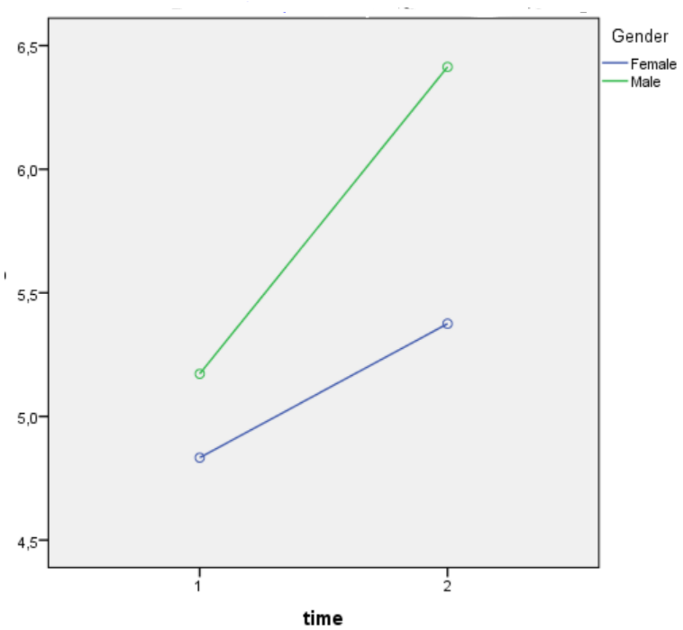


Figure 8. Estimated marginal means of Bebras scores

4.4 Difference in self-evaluation of CT skills between female and male students

Research Question 4: Is there a significant difference between male and female students' scores in terms of the CT Scale at the end of the treatment?

Students were asked to rate levels of agreement (from 1 – "I strongly disagree" to 5- "strongly agree") to the questions in the Korkmaz and colleagues' CT Scale. To analyze the data, descriptive and inferential statistics were used.

First, the data were checked for normal distribution. According to the results being showed in the Table 21 and the Table 22, the data were normally distributed.

Descriptive statistics indicated that female students' mean of the CT Scale scores was 85.75 (*SD* = 16.39) and male students' mean of the CT Scale scores was 79.34 (*SD* = 15.17) (see Table 21). That is, female students evaluated themselves with higher ranks in CT than the male students.

Table 21. Descriptive Statistics for Students' Scores in Terms of the CT Scale

		N	Skewness	Kurtosis	Min	Max	Mean	Median	SD
CT Scale	Female	24	-.99	0.63	42	106	85.75	90	16.39
	Male	29	-.55	-.49	46	101	79.34	80	15.17

Table 22. Shapiro-Wilk Test

		Statistic	df	Sig.
CT Scale	Female	.918	24	.053
	Male	.952	29	.203

Because the data was normally distributed independent sample t-test was applied. The Levene's test showed that the assumption of homogeneity of variance was met for the CT scale scores, $p = .724 > .05$. The results showed there is no significant difference between female ($M = 85.75$, $SD = 16.39$) and male ($M = 79.34$, $SD = 15.17$) students' scores in terms of the CT Scale, $t(51) = 1.47$, $p = .146$ (see Table 23).

Table 23. Independent Samples T- Test

	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	95% Confidence Interval of the Difference Lower	Upper
Eql. variances assumed	1.47	51	.146	6.40	4.34	-2.31	15.12
Eql. variances not assumed	1.46	47.53	.150	6.40	4.37	-2.39	15.20

CHAPTER 5

DISCUSSION AND CONCLUSIONS

The present study examined the role of using unplugged computing activities on the development of middle school students' CT skills. Fifty-three 6th grade students participated in the study, and ten unplugged computing activities were used as class activities. These tasks were selected from Bebras challenges, revised and translated into Turkish by taking expert opinions. Each question was coded with one or more CT skill categories (abstraction, decomposition, algorithmic thinking and generalization) as defined by Selby and Woollard (2013). Further, tasks were selected considering three difficulty levels. Four of them were selected from easy category, three of them were selected from medium category and the other three of them were selected from difficult category based on Bebras categorization.

Using the same CT skill categories and difficulty levels two parallel tests were prepared to assess CT skills of participants. Questions in the tests were also selected from Bebras challenge and translated into Turkish by taking expert opinion. These parallel tests included questions from three different difficulty levels (5 easy, 5 medium, and 5 difficult), and each question in each group were assigned to one or more CT skills being defined by Selby and Woollard (2013) as mentioned before. The present study is important because there is lack of instructional materials for developing CT skills using unplugged computing activities in Turkish curriculum. We compiled ten unplugged computing activities with three different difficulty levels covering a range of CT components based on Bebras challenges and prepared explanation sheets for each activity for classroom use. Also, we prepared two equivalent tests including three difficulty levels and covering the same amount of CT components as in the activities in order to assess the development of students' CT skills. All of these can be used for 6th grade students by teachers teaching CT skills.

The findings of the study showed that there was a significant improvement in students' CT skills after participating in the unplugged computing instruction. Also, although participants' CT skills differed according to their gender, this result was not statistically significant. We have also seen that students' success levels changed regarding different difficulty levels based on a descriptive analysis. In easy and medium levels questions although male students were more successful than female students, in difficult level questions female students showed more improvement than male students. And no interaction effect was found between students' scores and their gender. In addition, there was no significant difference between female and male students in terms of Korkmaz and colleagues' CT Scale.

This chapter discusses the findings based on research questions, limitations of the study and makes suggestions for future research.

5.1 Unplugged computing instruction effect

For the first research question (is there a significant difference between students' Bebras test scores before and after attending to the unplugged computing instruction?), the results showed that there was a significant difference between students' pre and post test scores after attending the instruction. In other words, using unplugged computing activities developed students' CT skills.

Lu and Fletcher (2009) asserted that CT can be isolated from programming and should be taught before programming instruction starts. In the current study, CT is not taught using actual computer programming, and it was aimed to be developed by using unplugged computing activities. As stated in the literature section, there exists several unplugged computing projects (Bell, Alexander, Freeman, & Grimley, 2009; Curzon at al., 2009; Mittermeir, Bischof ,& Hodnigg, 2010 cited as Kalelioğlu, 2018; Gallenbacher, 2012 cited as Kalelioğlu, 2018) and Bebras (Bilge Kunduz*)* is one of them (https://www.bebras.org/) (Kalelioğlu, 2018). Bebras tasks were used in the current study and results showed that they developed participants' CT skills. For Bebras tasks, Dagiene and Stupuriene (2016, p. 40) stated that "the Bebras tasks focus on informatics concepts, support understanding of informatics phenomena and develop computational thinking". The present study is important because there is lack of instructional materials for developing CT skills using unplugged computing activities in Turkish curriculum. We compiled ten unplugged computing activities with three different difficulty levels enclosing a number of CT components based on Bebras challenges and prepared explanation sheets for each activity for classroom use. We also prepared two equivalent tests including three difficulty levels and covering the same amount of CT components as in the activities in order to assess the development of students' CT skills.

Even though unplugged computing activities have been presented to be successful in increasing students' interests, still less research is available on adapting unplugged computing activities for use in a classroom setting (Rodriguez, Rader & Camp, 2016). Moreover, Thies and Vahrenhold (2013) stated that unplugged computing activities were as successful as conventional approaches in introduction to CS concepts and algorithms to students. The findings of this study were consistent with the literature.

5.2 Gender effect

For the second research question (is there a significant difference between male and female students' Bebras test scores?), the results showed that there was no significant difference between students' Bebras test scores in terms of gender. Our findings are more in line with the results of the study of Kalaš and Tomcsányiová (2009) who also found that there were no significant differences in the performance of boys and girls. Atmatzidou and Demetriadis (2015) claimed that age and gender relevant differences occur when evaluating students' score in the various specific dimensions of the CT skills model. Their CT skills model included abstraction, generalization, algorithm, modularity and decomposition. In the light of this study, although it was not significant, gender relevant differences appeared when analyzing students' scores by considering questions' difficulty levels. Although in each group of questions, students increased their scores, in medium level questions female students showed a slight decrease in scores. While in easy questions, male students performed better than female students, in difficult questions girls were more successful than boys. This result was unexpected for us because when we looked at the literature it was stated that as questions' difficulty level increases, the level of female students' success decreases (Gülbahar, Kalelioğlu, & Doğan, 2016). Such a result may indicate that male students got bored at the end of the tests and difficult tasks required more attention than others. For female students, as the researcher observed them solving questions, they did not follow the questions' sequence linearly. They selected one question and tried to solve it. Thus, they may have spent more time on difficult questions getting more points.

Further, in two questions (questions 6 & 9 in the post-test) boys scored significantly higher than females. Question 9 (see Figure 9) may be more interesting for boys because it was about a running tournament. This may support Izu and colleagues (2016) argument, which was that boys performed better than girls in some tasks that required spatial abilities and depended on participants having an intuitive understanding.

9. Kunduz Turnuvası

Kunduz Kamil bir koşu turnuvasına izleyici olarak katılıyor. Turnuva boyunca her aşamadaki kazananları bir tabloya kaydediyor. Yarışmadaki koşucuların turnuva boyunca giydikleri forma numaraları değişmiyor. Kamil koşucuları temsil etmek için 1'den 8'e kadar numaralandırdığı kartları kullanıyor. Turnuva sonunda Kamil'in küçük kardeşi Taha ilk aşamadaki yarışma sonuçları hariç diğer aşamalardaki tüm kartları karıştırıyor.

Soru: Aşağıdaki boş kutulara yandaki kartlardan hangileri gelmelidir. Üzerlerine yaz.

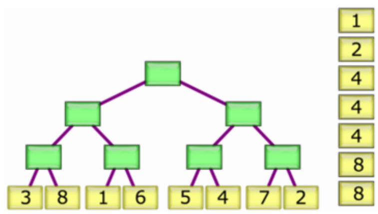


Figure 9. Question 9 in the post-test

5.3 Interaction effect

For the third research question (does any interaction occur between participants' gender (male and female) and the time of pre- and post-tests?), there was no interaction between students' pre and post test scores in terms of their gender. Students' gender did not affect the overall improvement in CT scores.

5.4 Difference in self-evaluation of CT skills between male and female students

For the last research question (is there a significant difference between male and female students' scores in terms of the CT Scale at the end of the treatment?), although the descriptive statistic results indicated that female students mean score was higher than male students' mean score, inferential statistic results showed that there was no significant difference between boys' and girls' scores in terms of the CT Scale. This result is consistent with the results of Korkmaz and colleagues (2015). They found no difference between male and female students' scores except critical thinking group questions. In the current study, the scale was given only after the treatment because the time interval between the start and the end time of the treatment was short.

5.5 Suggestions for future research

The present study is important because there is lack of instructional materials for developing CT skills using unplugged computing activities in Turkish curriculum. We compiled ten unplugged computing activities with three different difficulty levels enclosing a range of CT components based on Bebras challenges and prepared explanation sheets for each activity for classroom use. However, more research is needed on the role of unplugged computing activities in developing CT skills.

In this study, the CT definition of Selby and Woollard was taken into consideration. They defined CT as a skill which includes abstraction, decomposition, algorithmic thinking, generalization and evaluation (Selby & Woollard, 2013). However, in the current study we selected four CT skills from their CT definition, which are abstraction, decomposition, algorithmic thinking and generalization. For further research, a more complex CT definition that includes more CT aspects may be made, and more CT components may be selected; in turn, more questions may be developed.

Furthermore, in the current study unplugged computing activities were used and it was applied in three class hours. In further research the application time may be extended. Thus, the role of unplugged computing activities on developing CT skills can be observed better. As being stated in the study of Atmatzidou and Demetriadis (2015), CT skills in most cases need time to fully develop. Also, in these activities aiming to develop CT skills, students learned and used skills related to abstraction, decomposition, algorithmic thinking and generalization. Further research can investigate whether these CT skills can transfer in other areas. Thus, researchers can better judge whether unplugged computing activities are useful in other disciplinary areas.

APPENDIX A

AN EXAMPLE EXPLANATION SHEET

FOR BEAVERS ON THE RUN ACTIVITY

Story: Five beavers who were labeled as A, B, C, D, E that went to a forest for a walk and there were four different depth pits on their way. Four beavers fit into in the first pit, two beavers fit into in the second pit, and three beavers fit into in the third pit. Blue lines on the Figure 2 showed how many beavers could fit into the pit. Beaver has been given the rules to cross the pits. According to this rule, what order do beavers find themselves after they passed third hole? (Question mark on the Figure 2 showed final position place).

Difficulty level: Easy

CT skill: Abstraction, Decomposition, Algorithmic thinking

This activity is an example of data structures that have an important place in computer science. In the stack data structure, you always add one to the top, and the first element is removed from the top. This is called LIFO. Most recently added first subtracted.

Teacher role:

Before starting the main activity, the instructor is asked to go over a mini activity about beavers who walked into the forest (see Figure 1) and students discuss their solutions. In this mini activity, beavers went out for a walk in the woods. The roads in the jungle are narrow, so the beavers can move in one row. In some places there are pits, there are certain rules to pass these pits.

1. First as many beavers jump into the hole as fit in.
2. The entire colony will then pass across the hole.

3. The beavers that jumped in will then climb out.

The order of the beavers in the forest and the rules are projected onto the blackboard (see Figure 1). They discuss the beavers' positions after they passed the pit. Participation of all students is ensured (5min).

After the discussion, the sheet of "Beavers on the Run" is given to students (see Figure 2) and the instructor announces the time for this activity.

The students are asked to rank the beavers (10-12 min). Then, students start to solve the activity by themselves. After they finished, they discuss their solutions and one is asked to explain his/her solution to the whole class.

Rules:

1. First as many beavers jump into the hole as fit in.
2. The entire colony will then pass across the hole.
3. The beavers that jumped in will then climb out.

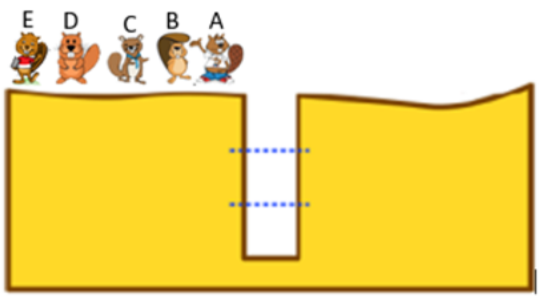


Figure 1. Mini activity for the order of beavers in forest

Five beavers who were labeled as A, B, C, D, E that went to a forest for a walk and there were four different depth pits on their way. Four beavers fit into in the first pit, two beavers fit into in the second pit, and three beavers fit into in the third pit. Blue lines on the Figure 2 showed how many beavers could fit into the pit. Beaver has been given the rules to cross the pits. According to this rule, what order do beavers find themselves after they passed third hole? (Question mark on the Figure 4 showed final position place).

Rules:

1. First as many beavers jump into the hole as fit in.
2. The entire colony will then pass across the hole.
3. The beavers that jumped in will then climb out.

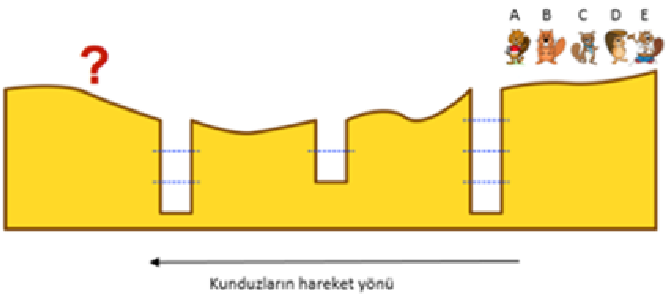


Figure 2. Main activity- Beavers on the run

AN EXAMPLE EXPLANATION SHEET

FOR BEAVERS ON THE RUN ACTIVITY (TURKISH)

Ormanda Yürüyüş

Hikaye: A, B, C, D, E olarak adlandırılan 5 kunduz, ormana yürümeye gitmiş. İlerledikleri yolda 4 tane farklı derinlikte çukur var. İlk çukura 4 kunduz, 2. çukura 2 kunduz, 3. çukura 3 kunduz sığabiliyor. Şekil 2 üzerinde verilen mavi çizgiler çukura kaç kunduz inebileceğini gösterir. Kunduzların çukurları geçme kuralları verilmiş. Bu kurala göre yolun sonunda kunduzların sırası nasıl olur? (Şekil 2'deki soru işareti kunduzların son konumlarının olacağı yeri gösterir.)

Zorluk Seviyesi: Kolay

CT Kavramları: *Soyutlama, Ayrıştırma, Algoritmik Düşünme*

Bu etkinlik bilgisayar biliminde önemli bir yere sahip olan veri yapılarından yığına (stack) bir örnek niteliğindedir. Yığın veri yapısında hep en üste bir tane eklersiniz ve ilk eleman çıkartılacağı zaman en üstten çıkartılır. Bu LIFO diye adlandırılır. En son eklenen ilk çıkartılır şeklinde

Öğretmen:

Ana etkinliğe geçmeden önce öğretmen öğrencilerle birlikte küçük bir etkinlik yapar (bkz. Şekil 1). Bu etkinlikte, kunduzlar ormanda yürüyüşe çıkmış. Ormanda yollar dar ve bu yüzden tek sıra yürümek zorundalar. Bazı yerlerde ise çukurlar var. Bu çukurları geçmek için belirli kurallar verilmiş.

1. Kaç tane kunduz çukura inebilirse iner.

2. Geri kalan kunduzlar çukurdan karşıya geçer.

3. Çukurdaki kunduzlarda çıkıp yola devam eder.

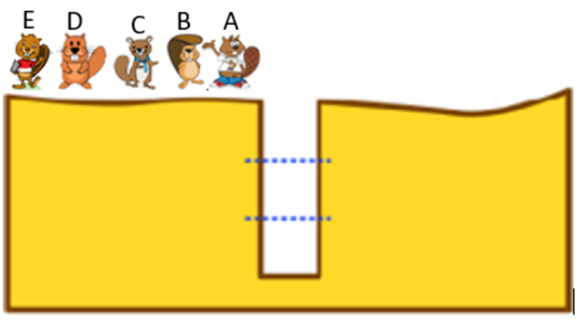
Kunduzların ormandaki sırası ve kurallar projeksiyonla tahtaya yansıtılır (bkz. Şekil 1). Kunduzların çukuru geçtikten sonra ki konumları tartışılır. Tüm öğrencilerin katılımı sağlanır (5dk).

Tartışma bittikten sonra, ana etkinlik kağıdı “Ormanda Yürüyüş” öğrencilere dağıtılır (bkz. Şekil 2) ve öğretmen bu etkinlik için belirlene süreyi söyler (10-12dk).

Öğrencilerden kunduzları 10-12 dakika içerisinde verilen kurallara göre sıralamaları istenir. Öğrenciler etkinliği kendi başlarına çözmeye başlar. Süre bittikten sonra çözümlerini tartışırlar ve içlerinden birinden çözümünü tüm sınıfa anlatması istenir.

Kurallar:

1. Kaç tane kunduz çukura inebilirse iner.
2. Geri kalan kunduzlar çukurdan karşıya geçer.
3. Çukurdaki kunduzlarda çıkıp yola devam eder.



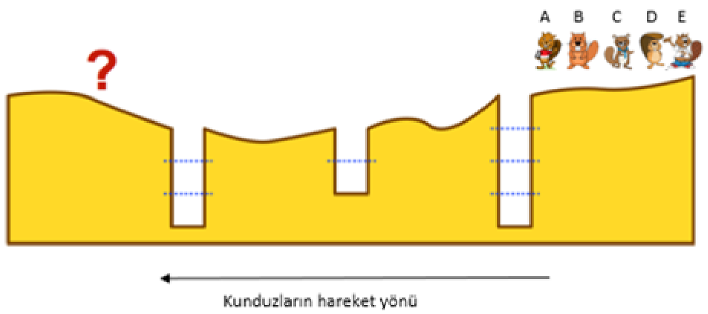
Şekil 1. Küçük etkinlik- kunduzların ormandaki sırası

Ormanda Yürüyüş

A, B, C, D, E olarak adlandırılan 5 kunduz, ormana yürümeye gitmiş. İlerledikleri yolda 4 tane farklı derinlikte çukur var. İlk çukura 4 kunduz, 2. çukura 2 kunduz, 3. çukura 3 kunduz sığabiliyor. Şekil 2 üzerinde verilen mavi çizgiler çukura kaç kunduz inebileceğini gösterir. Kunduzların çukurları geçme kuralları verilmiş. Bu kurala göre yolun sonunda kunduzların sırası nasıl olur? (Şekil 2'deki soru işareti kunduzların son konumlarının olacağı yeri gösterir.)

Kurallar:

1. Kaç tane kunduz çukura inebilirse iner.
2. Geri kalan kunduzlar çukurdan karşıya geçer.
3. Çukurdaki kunduzlarda çıkıp yola devam eder.



Şekil 2. Ana etkinlik- ormanda yürüyüş

APPENDIX B

PRE-TEST

1. Only nine keys

Daniel is sending text messages from his old phone.

For every letter he has to press the proper key once, twice, three or four times, followed by a short pause.

In order to type 'C' he has to press the number 2 key three times because 'C' is the third letter written on this key.

In order to type 'HIM' he has to press the number 4 key twice, followed by the number 4 key 3 times and finally the number 6 key once.

Daniel presses exactly six times to enter the name of a friend.

What is the name of his friend?

Miriam

Iris

Emma

Ina

2. Message service

Violet wants to send a message to Leo with the help of some beavers and some cards.

She breaks the message into groups of, at most, 3 letters on each card.
She then gives one card to each beaver.

Violet knows that sometimes the beavers get distracted while transporting their cards, and they arrive at different times. Therefore, Violet also numbers the cards in the correct order before giving them to the beavers. Leo must then put then back in order to read the message.

Example:
To send the message DANCETIME, Violet creates these 3 cards:

1 DAN	2 CET	3 IME

Leo received the following sequence of cards from the beavers:

3 KEY	5 CKS	2 HOC	1 GET	4 STI

Question:
What was the original message that Violet sent to Leo?

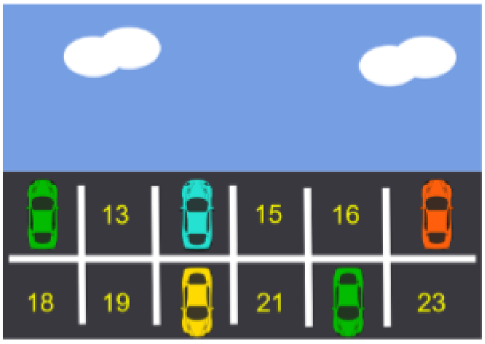
GETSTICKYSHOCKS STICKYGETHOOKS GETHOCKEYSTICKS KEYCKSHOCGETSTI

3. Parking lot

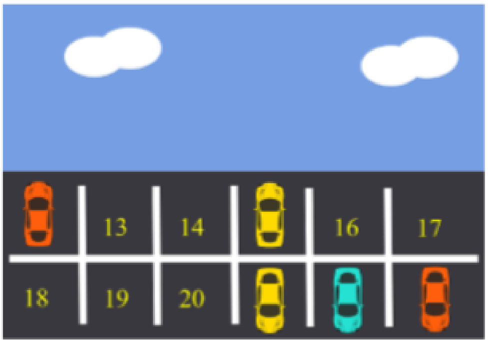
There are 12 spaces for cars in a parking lot. Each space is labelled with a number.

The pictures below show which spaces were used on Monday and which spaces were used on Tuesday.

Monday



Tuesday



Question:

How many parking spaces were empty both on Monday and Tuesday?

3 4 5 or 6

4. Sticks and shields

Lucia is playing Stick and Shield with 7 friends.

These are her friends' favourite poses:

They want to have their picture taken.

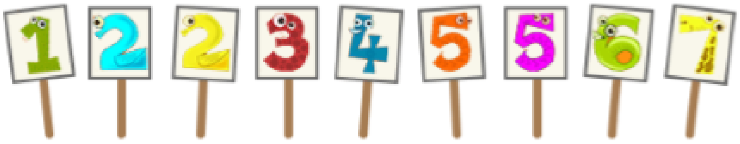
In the picture every stick should point at another beaver, and every shield should block a stick.

Lucia has already taken a spot ready for the picture.

Task:

Drag the friends, shown below, into their correct positions.

5. Animal competition



The beavers and dogs had a competition. In total nine animals took part.

The nine participants had the following scores: 1, 2, 2, 3, 4, 5, 5, 6, 7.

No dog scored more than any beaver.

One dog tied with a beaver.

There were also two other dogs that tied with each other.

Question:

How many dogs took part in the competition?

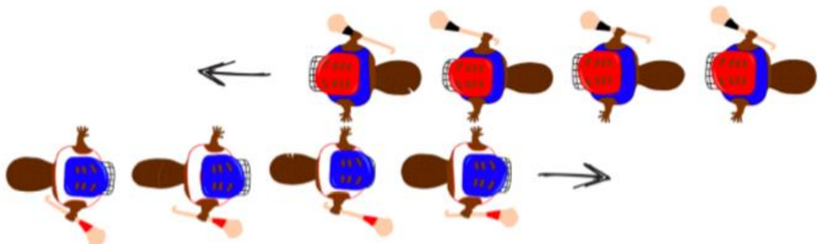
2, 3, 5, 6 or 7

6. Hurlers shake hands

Beavers enjoy playing hurling.

After the game ends, the beavers in each of the two teams line up in a row and walk past the other team. As they pass each other, they shake hands. At the beginning, only the first player on each team shakes hands. Next, the first two players shake hands (see picture below). This continues until each player has shaken hands with every player on the other team.

There are 15 players on each team.



Question:

If each player takes one second to shake hands and move to the next player, how many seconds of shaking hands will there be?

7. Candy maze

A robot is programmed to collect as many sweets as possible. It does this while walking through cells. Each cell in the grid on the right has either 0, 1, 2 or 3 sweets.

The robot begins in the bottom-left and ends in the top-right. The robot can only move to the right or upwards.

Question:

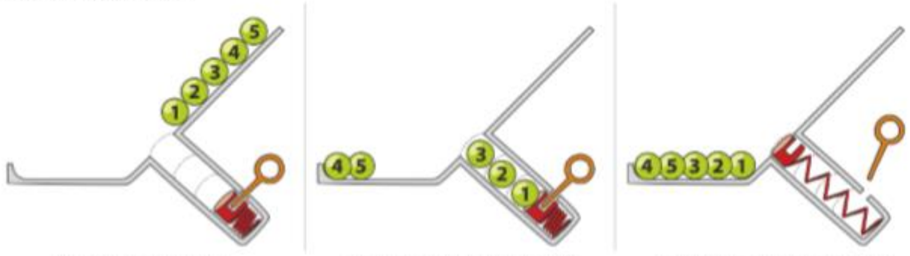
How many sweets will the robot collect in this grid?

10 12 14 or 16

8. Balls

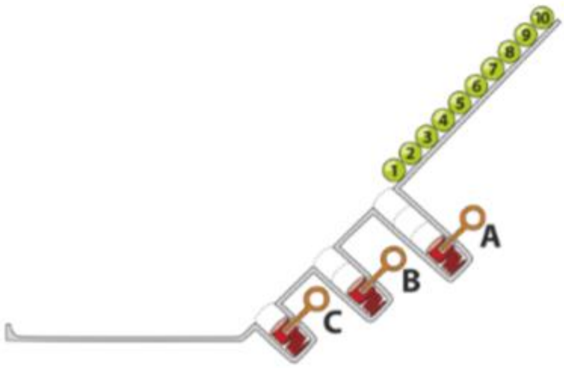
Numbered balls roll down ramps. The order of the balls changes as they fall into holes. When a ball comes to a hole, if there is enough space, the ball falls in, otherwise, the ball rolls past the hole. A pin at the bottom of each hole can be pulled which ejects the balls.

Here is an example:

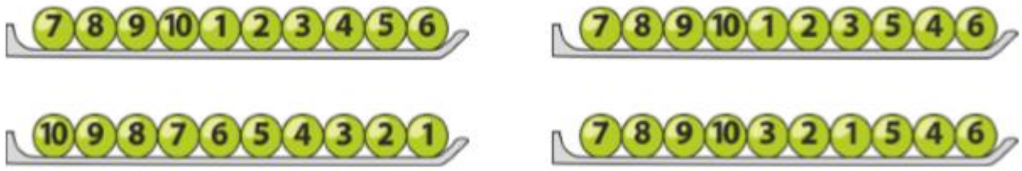


before balls start rolling | after balls have stopped rolling | final result after pulling the pin

Ten balls roll down the ramp shown below.
Three holes A, B and C have space for 3, 2 and 1 balls as shown.
The pins are pulled in the order A, B, C but, each time, only after all the balls have stopped rolling.



Question:
Which of the following is the final result?



9. Setting the table

Beaver Bob has set the breakfast-table as shown in the picture.

Question:

In which order has he placed the objects on the table?

A. table cloth, napkin, cup and saucer, knife, plate

B. table cloth, napkin, cup and saucer, plate, knife

C. napkin, knife, table cloth, cup and saucer, plate

D. table cloth, cup and saucer, napkin, plate, knife

10. Levenshtein distance

We define a basic operation as one of the following:

- insert one character into a word,
- remove one character from a word,
- replace one character with another.

We define the distance between two words as the minimum number of basic operations which allows us to change the first word into the second.

For example, the distance between kitten and sitting is equal to 3. The basic operations necessary are:

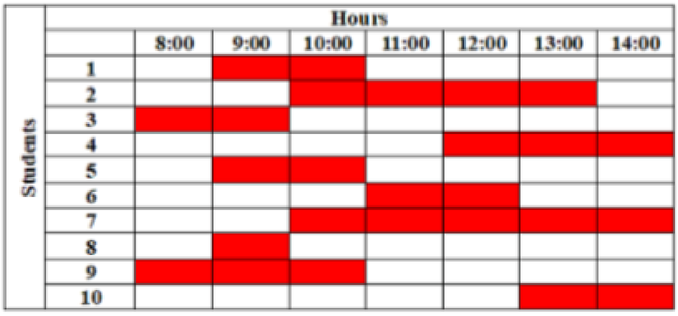
1. kitten → sitten (change k to s),
2. sitten → sittin (change i to e),
3. sittin → sitting (insert g at the end).

Question:
What is the minimum distance between length and french?

11. News editing

There are 10 students working on the school's newspaper. Every Friday they write or edit their own articles.

The red cells, on the plan below, show when the students need a computer.
The computers are all the same.
During any one hour, only one student at a time can work on a computer.



Question:
What is the minimum number of computers needed for all of the students to work according to the plan shown above?

4 5 6 or 10

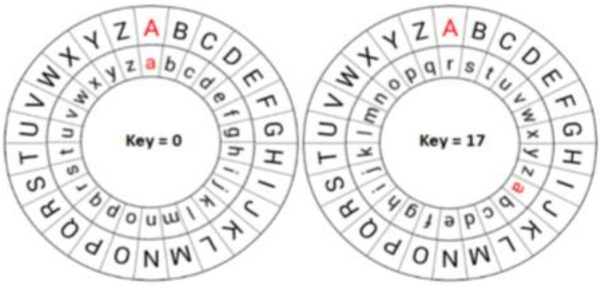
12. Chipher Wheel

A beaver left a secret message on his tombstone by using a cipher wheel and we want to figure out what it means.

The wheel works such that only the inner wheel (with small letters) can be rotated. The outer wheel is for the actual message.

As you can see in the first image, when the key is 0 'A' is encoded as 'a'.

The second image shows that when the key is 17 (because the inner wheel has been rotated by 17 positions counter-clockwise) 'A' is encoded as 'r'.



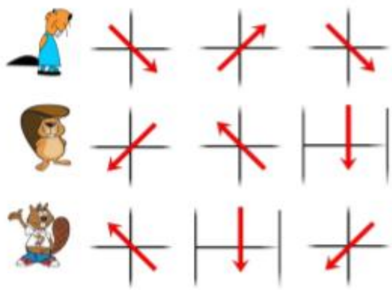
With the key equal to 17, we can encode the message WHO ARE YOU as nyf riv pfl

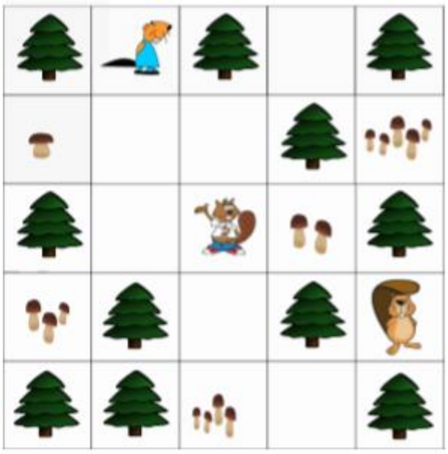
The message j cp fjgcma is received. We know that this was encrypted in a clever way:
For the first letter the key was 1, for the second letter the key was 2, the key for the third letter was 3, etc.

Question:
Decipher the encrypted message and enter the original message as your answer.

13. Mushrooms

Three beavers are standing in a forest.
Each wants to go where there are mushrooms.
Arrows in the picture to the right show the directions the beavers will walk.



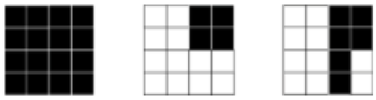


Question:

Where do the beavers end up?

14. Icon image reduction

Look at the following 4x4 black and white pixel images:



This could be stored using binary digits: “1” for white pixels and “0” for black pixels.
For a 4x4 image we would have to store 16 digits. The following image compression method allows us to store images using less space, especially for simple patterns:

0 0 0 0	1 1 0 0	1 1 0 0
0 0 0 0	1 1 0 0	1 1 0 0
0 0 0 0	1 1 1 1	1 1 0 1
0 0 0 0	1 1 1 1	1 1 0 1
0	(1011)	(10(0110)1)

The binary digits are arranged in a grid like the pixels in the images.

The compression method is applied to this grid as follows, producing a string of digits:

1. If all the digits in the grid are 0, the result is ”0” (see left image).
   If all the digits in the grid are 1, the result is ”1”.
2. Otherwise, the grid is divided into quarters. The compression method is applied to each quarter sub-grid from the top left in clockwise order. The results are combined and surrounded by round brackets. Two different examples can be seen in the centre and on the right above.

Note that a sub-grid may consist of one digit only; see the right image, bottom right corner.
In this case, the method will use step 1 only.

Question:

On the right is the binary digit grid for an 8x8 image.
The above compression method is applied to this grid.
Which string of digits can represent this image?

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

(1110)

(11(1011)1)

(111(1(1101)11))

(111(1(1011)11))

15. Stack computer

The Stack Computer is loaded with boxes from a conveyer belt. The boxes are marked with a Number or an Operator (+, -, * or /).

The computer is loaded until the top box is a box marked with an operator. This operator is then used on the two boxes below it. The three boxes are then fused into one single box and marked with the outcome of the calculation.

In the Stack Computer, calculations are entered in a different way to a normal calculator.

Examples:

2+3 must be entered as 2 3 +

10-2 must be entered as 10 2 -

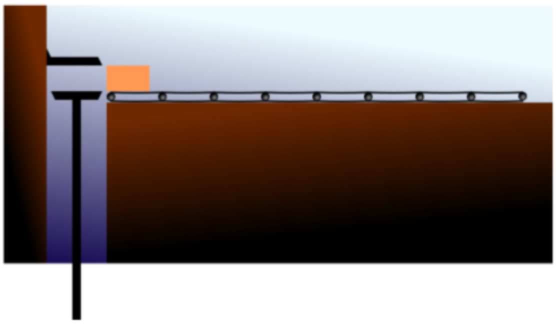
5*2+3 must be entered as 5 2 * 3 +

5+2*3 must be entered as 5 2 3 * +

(8-2)*(3+4) must be entered as 8 2 - 3 4 + *

Question:

How should the following computation be entered: 4*(8+3)-2?



PRE-TEST (TURKISH)

1. Telefon Tuşları

Ali, yandaki şekildeki gibi tuşları olan bir mobil telefon ile arkadaşına, bir ismi mesaj olarak göndermek istemektedir. Aynı tuşa birkaç kez basıldığında harfler ekranda gözükmektedir.

Örneğin, C harfi için  tuşuna 3 kere basmak gerekmektedir.
CAN kelimesini yazmak için ise  tuşuna 3 kere, sonra  tuşuna bir kere ve son olarak  tuşuna iki kere basılmalıdır.

Soru: Ali, 7 kez tuşa basarak arkadaşının ismini mesaj olarak göndermiştir. Aşağıdaki seçeneklerden hangisi gönderdiği isimdir?

A)KAYA B) IRMAK C) MELEK D) TUFAN

---

2. Mesaj İletimi

Ayşe arkadaşı Can'a kunduzların yardımı ile uzun bir mesaj göndermek ister. Ayşe, mesajları her bir kartta en fazla 3 harf olacak şekilde gruplara ayırır ve bu kartların her birini bir kunduza verir. Mesaj taşıyan kunduzlar kartları Can'a farklı zamanlarda iletmektedir. Bu yüzden, Ayşe kartları kunduzlara vermeden önce her birine sıra ile numara verir. Can, mesajı anlayabilmek için kartları sıralamak zorundadır. Örneğin, KELİMEBUL mesajı için, Ayşe'nin oluşturduğu 3 kart aşağıdaki gibidir.

1 KEL	**2** İME	**3** BUL

Can aşağıdaki mesajı almıştır.

3 İŞB	**5** NDA	**2** DUZ	**1** KUN	**4** AŞI

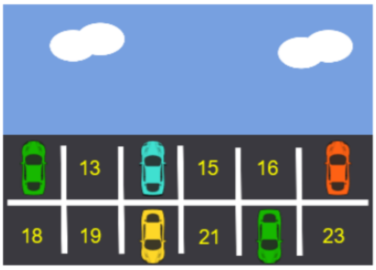
Soru: Can'ın aldığı mesaj nedir?

A) NDADUZİŞBKUNAŞI
B) İŞBAŞINDAKUNDUZ
C) KUNDUZİŞBAŞINDA
D) KUNDUZAŞINDAİŞB

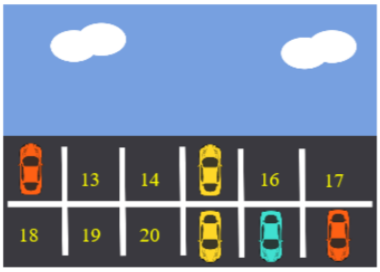
3. Otopark

Bir otoparkta toplam 12 boş park yeri bulunmaktadır. Her yerin belirli bir numarası vardır. Aşağıda verilen resimler Pazartesi ve Salı günleri hangi yerlerin kullanıldığını göstermektedir.

Pazartesi günü otoparkın durumu



Salı günü otoparkın durumu



Soru: Hem Pazartesi hem de Salı günü otoparkta toplam kaç boş yer vardır?

A) 3 B) 4 C) 5 D) 6

---

4. Kılıç Kalkan Oyunu

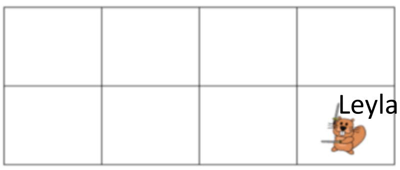
Kunduz Leyla, 7 arkadaşıyla kılıç kalkan oynuyor.

Bunlar arkadaşlarının favori pozları:

Leyla ve arkadaşları fotoğraflarının çekilmesini istiyorlar. Resimde her kılıç başka bir kunduza işaret etmeli ve her kalkan bir kılıcı engellemelidir. Leyla zaten fotoğraf için hazır bir yer buldu.

Soru: Leyla'nın arkadaşları verilen kurala göre hangi karelerde durmaları gerektiğini bilmiyorlar. Leyla'nın arkadaşlarının adlarını uygun karelere yaz.

Fotoğraf kareleri



5. Hayvanlar Yarışıyor

Kunduzlar ve köpekler bir yarışma yapmışlar. Bu yarışmaya toplam 9 hayvan katılmış. Bu 9 hayvanın yarışmadan kazandığı puanlar ise şöyle: 1, 2, 2, 3, 4, 5, 5, 6, 7

- Hiçbir köpek kunduzlardan daha yüksek puan alamamış.
- Bir kunduz bir köpekle aynı puanı almış.
- Köpeklerden iki tanesi de aynı puanı almış.

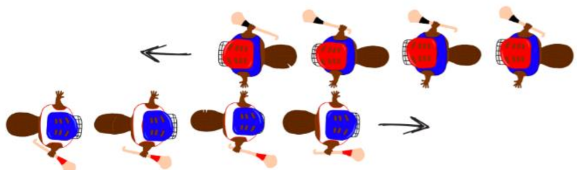
Soru: Verilen bilgilere göre bu yarışmada kaç köpek yer almıştır?

A) 2 B) 3 C) 5 D) 6

---

6. Buz Hokeyi

Kunduzlar buz hokeyi oynamayı seviyor. Oyun bittikten sonra, iki takımın her birindeki kunduzlar art arda dizilir ve diğer takımın yanından geçer. Birbirlerini geçtiklerinde, el sıkışırlar. Başlangıçta, sadece her takımdaki ilk oyuncu el sıkışır. Ardından, ilk iki oyuncu el sıkışır (aşağıdaki resme bakın). Bu, her oyuncu diğer takımdaki her oyuncu ile el sıkışıncaya kadar devam eder. Her takımda 15 oyuncu var.



Soru: Her oyuncunun el sıkışıp bir sonraki oyuncuya geçmesi bir saniye sürerse, tüm el sıkışma kaç saniye sürer?

A) 30 B) 29 C) 28 D) 27

---

7. Şeker Labirenti

Bir robot, karelerden oluşan bir alanda ilerleyerek mümkün olduğu kadar çok şeker toplayacak şekilde programlanmıştır. Aşağıdaki şekildeki her bir karede 0, 1, 2 veya 3 adet şeker vardır. Robot sol alt köşedeki kareden hareketine başlayıp sağ en üst köşedeki karede bitirir. Robot sadece sağa veya yukarı doğru hareket edebilir.

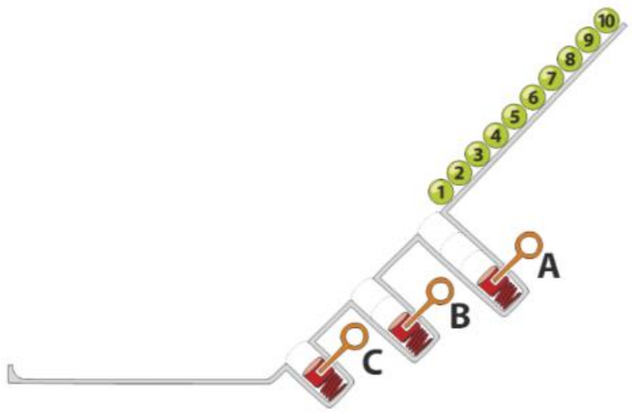
Soru: Bu robot yukarıdaki karelerden oluşan bir alanda kaç tane şeker toplar?

A) 10 B) 12 C) 14 D) 16

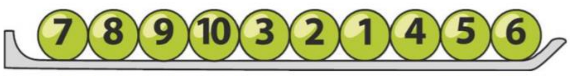
8. Bilyeler

Numaralı bilyeler rampadan yuvarlanmaktadır. Bilyelerin sırası kanallara düşerken değişmektedir. Bir bilye kanala geldiği zaman, eğer yeterince yer varsa içeri düşer, yoksa ileriye doğru yuvarlanır. Her kanalın altında bilyeleri geri iten bir yay bulunmaktadır.

On tane bilye rampadan yuvarlanır. A, B ve C olmak üzere 3 tane kanal ve kanallarda sırayla 3, 2 ve 1 bilye için yer vardır. Yaylar önce A, sonra B ve en son C olacak şekilde sırayla çekilir. Ancak her yayı çekmeden önce diğer tüm bilyeler yuvarlanmış olmalıdır.



Soru: Buna göre son durum aşağıdakilerden hangisidir?

A) 

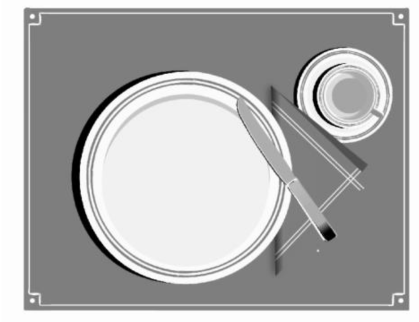
B) 

C) 

D) 

9. Sofra Düzeni

Bilge Kunduz ailesi için resimde görülen kahvaltı sofrasını hazırlamıştır.



Soru

Bilge Kunduz masanın üzerindeki nesneleri hangi sırada yerleştirmiştir?

A. Masa örtüsü, peçete, fincan, bıçak, tabak

B. Masa örtüsü, fincan, peçete, tabak, bıçak

C. Masa örtüsü, peçete, fincan, tabak, bıçak

D. Peçete, bıçak, masa örtüsü, fincan, tabak

---

10. Dizi Mesafesi

Aşağıda kelimelerle ilgili işlemler listelenmiştir.

- Kelimeye bir karakter ekleme
- Kelimeden bir karakter çıkarma
- Kelimedeki karakterlerin yerlerini değiştirme

İki kelime arasındaki fark, ilk kelimenin diğerine çevrilmesini sağlayan en az işlem sayısıdır. Örneğin, halk ve ulak kelimeleri arasındaki fark üçtür:

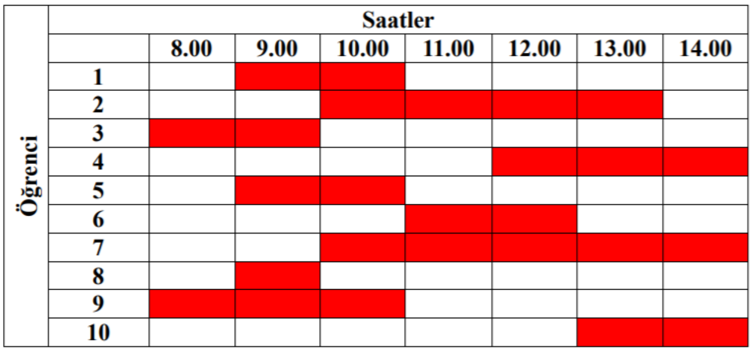
1. halk →hlak (a harfi l harfiyle değiştirilmiştir)
2. hlak → lak (h harfi çıkartılmıştır)
3. lak → Ulak (başa u harfi eklenmiştir)

Soru: Kalem ve elmas dizileri arasında olabilecek **en az** fark nedir?

A) 4 B) 5 C) 6 D) 7

11. Okul Gazetesi

Okul gazetesinde çalışmakta olan 10 öğrenci bulunmaktadır. Her Pazartesi bu öğrenciler köşe yazılarını yazmakta veya düzenlemektedir. Aşağıdaki tabloda boyalı hücreler öğrencilerin çalışmak için bilgisayara ihtiyaç duyduğu zamanları göstermektedir. Herhangi bir saat boyunca bir öğrenci sadece bir bilgisayarda çalışabilmektedir.



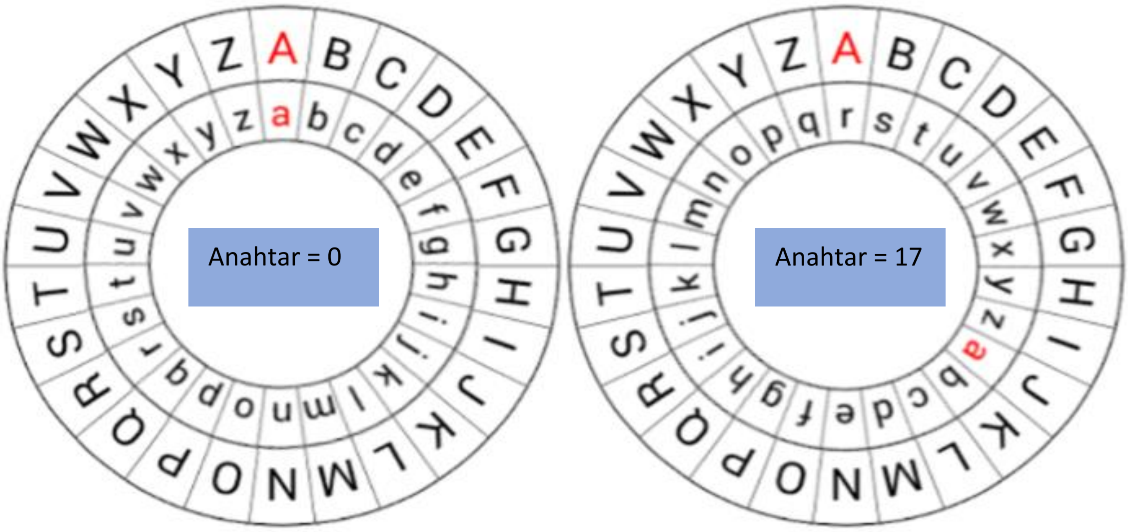
Soru: Öğrencilerin yukarıda belirtilen plana göre çalışabilmesi için gereken **en az** bilgisayar sayısı kaçtır?

A) 4 B) 5 C) 6 D) 10

12. Şifre Çarkı

Bir kunduz, bir şifre çarkı kullanarak mezar taşı üzerinde gizli bir mesaj bıraktı ve bunun ne anlama geldiğini öğrenmek istiyoruz. Tekerlek yalnızca iç tekerleğin (küçük harflerin bulunduğu tekerlek) döndürülebileceği şekilde çalışır. Dış tekerlek asıl mesaj içindir.

İlk resimde gördüğünüz gibi, anahtar 0 ‘A’ iken ‘a’ olarak kodlanır. İkinci görüntü, anahtar 17 iken (iç tekerleğin saat yönünün tersine 17 kere döndürüldüğü için) 'A' 'r' olarak kodlandığını gösterir.



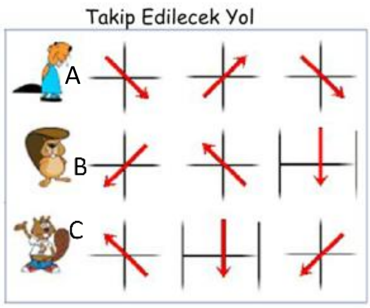
Anahtar 17 iken WHO ARE YOU mesajı nyf riv pfl olarak şifrelenmiştir.

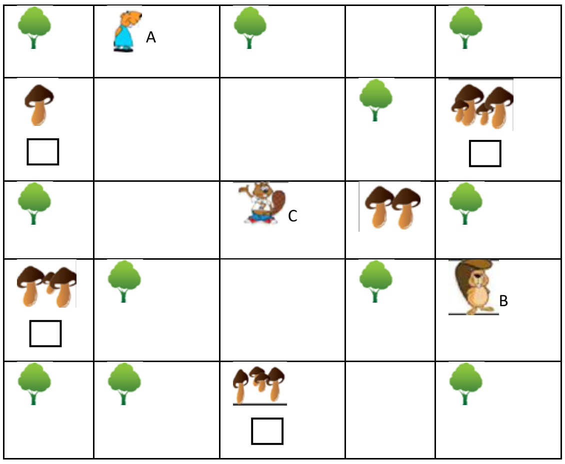
Mezar taşının üzerine **j cp fjgema** mesajı yazılmış. 1. harf için anahtar 1 kez, 2. harf için anahtar 2 kez, 3. harf için anahtar 3 kez çevrilmiştir. Diğer harfler içinde aynı şekilde devam eder.

Soru: Şifreli **j cp fjgema** mesajını deşifre edin ve orijinal mesajı yazın.

13. Mantar Toplama

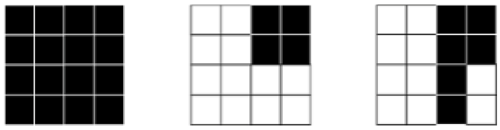
A, B, C diye adlandırılmış üç kunduz mantar toplamak için ormana gitmiş. Mantarlara gitmek için her kunduzun izlemesi gereken yol sağ tarafta verilmiş. Yolu takip ederek kunduzların adlarını gitmeleri gereken mantarların olduğu yerdeki etikete yaz.





14. Görüntü İşleme

Aşağıdaki 4x4 siyah beyaz piksel resimlere bakın:



Bu resimler, beyaz pikseller için “1”, siyah pikseller için “0” verilerek ikili rakamlar halinde kaydedilebilir. 4x4 görüntü için 16 basamak saklamamız gerekir. Aşağıdaki görüntü sıkıştırma yöntemi, özellikle basit kalıplar için görüntüleri daha az alan kullanarak depolamamıza izin verir:

0	0	0	0		1	1	0	0		1	1	0	0
0	0	0	0		1	1	0	0		1	1	0	0
0	0	0	0		1	1	1	1		1	1	0	1
0	0	0	0		1	1	1	1		1	1	0	1
0					(1011)					(10(0110)1)			

İkili rakamlar, görüntüdeki pikseller gibi bir ızgarada düzenlenir.

Sıkıştırma yöntemi, aşağıdaki gibi uygulanmakta ve bir basamak rakamı üretilmektedir:

1. Kılavuzdaki tüm rakamlar 0 ise, sonuç “0” dır (soldaki resme bakınız).
   Kılavuzdaki tüm rakamlar 1 ise, sonuç “1” olur.
2. Aksi halde, ızgara çeyreklere ayrılır. Sıkıştırma yöntemi her çeyrek alt ızgaraya sol üstten saat yönünde sırayla uygulanır. Sonuçlar bir araya getirilmiş ve yuvarlak parantez içinde çevrelenmiştir. Ortadaki ve sağdaki iki farklı örnekte bunu görebilirsiniz.

Not: Bir alt ızgaranın yalnızca bir rakamdan oluşabileceğini unutmayın. Sağdaki resmin sağ alt köşesinin ızgaradaki gösterimine bakın. Yöntem olarak yalnızca 1. kuralın uygulandığını göreceksiniz.

Soru: Sağda 8 × 8 görüntü için ikili basamaklı ızgaralı gösterimi yazılmış. Yukarıdaki sıkıştırma yöntemine göre bu ızgarayı gösteren rakam dizisi aşağıdakilerden hangisidir?

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

A) (1110)
B) (11(1011)1)
C) (111(1(1101)11))
D) (111(1(1011)11))

15. Paketleme Makinesi

Yürüyen bant üzerinden paketleme makinasına kutular geliyor. Gelen kutular bir sayı ya da operatörle (+, -, * veya /) etiketlenmiştir. Bu paketleme makinası operatörle işaretlenmiş bir kutu gelene kadar kutu almaya devam eder. Herhangi bir operatörle işaretli bir kutu geldiğinde ise operatörden önce gelen son iki kutunun üzerindeki sayıları, gelen operatörle işleme koyar. Daha sonra bu üç kutuyu tek bir pakete koyup üzerine operatörle yaptığı işlemin sonucunu yazar.

Paketleme makinesi normal hesap makinesinden farklı çalışır.

Örnek

Makinenin 2+3 işlemini yapması için, makineye 2 3 + şeklinde gelmiş olmalı

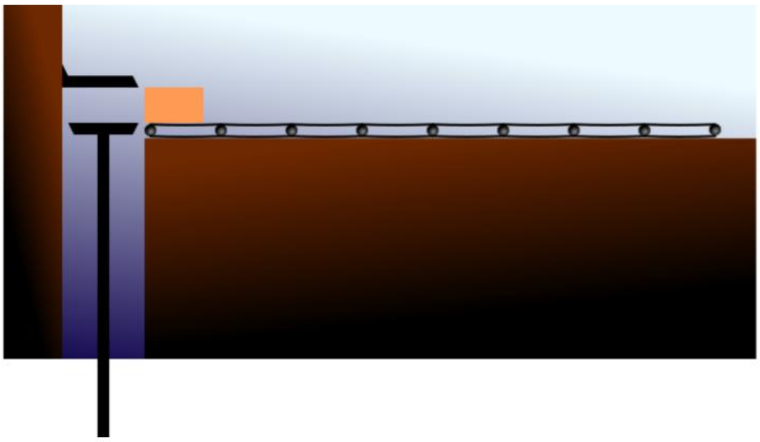
Makinenin 10-2 işlemini yapması için, makineye 10 2 – şeklinde gelmiş olmalı

Makinenin 5*2+3 işlemini yapması için, makineye 5 2 * 3 + şeklinde gelmiş olmalı

Makinenin 5+2*3 işlemini yapması için, makineyie2 3 * 5 + şeklinde gelmiş olmalı

Makinenin (8-2)*(3+4) işlemini yapması için, makineye 8 2 – 3 4 + *

Soru: Makinenin 4*(8+3)-2 bu işlemi yapması için makineye kutular hangi sırayla gelmelidir, aşağıya yaz.



APPENDIX C

POST-TEST

1. Ice Cream

At the LIFO ice cream parlour the scoops of ice cream are stacked on your cone in the exact order in which you ask for them.

What do you have to say in order to get the ice cream shown in the picture?

I would like to get an ice cream with ...

... Chocolate, Smurf and Strawberry!

... Strawberry, Smurf and Chocolate!

... Chocolate, Strawberry and Smurf!

... Strawberry, Chocolate and Smurf!

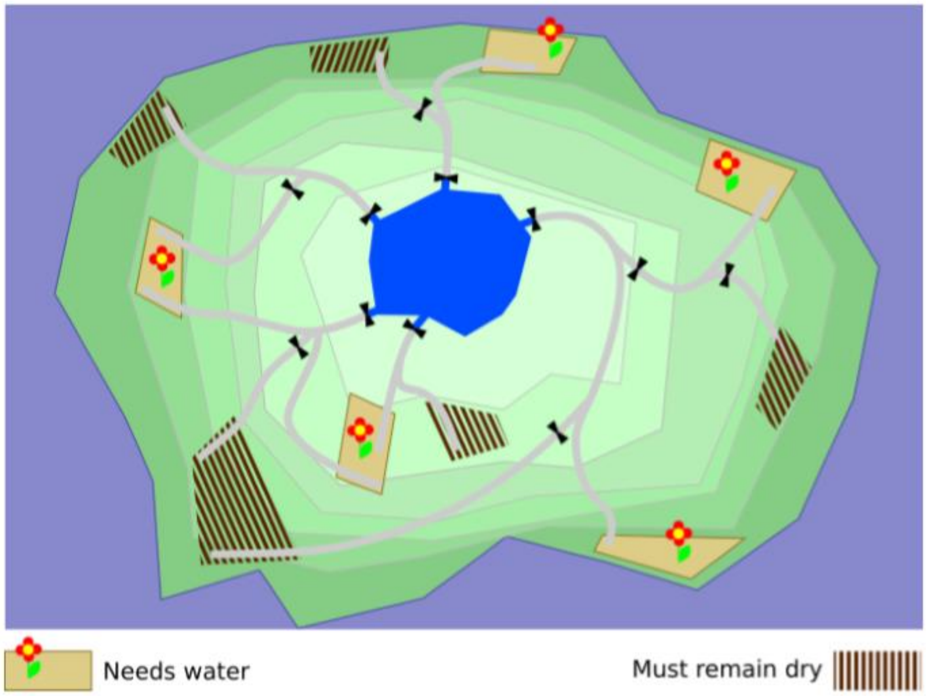
2. Beaver Gates

The Birchtree family needs to water their fields. Only fields with flowers need to be watered. The other fields must remain dry.

Click on a black gate to close or open it.

If it is open, water will flow from the lake in the middle to the fields below.

Help the Birchtree family produce a plan by clicking on the gates to open or close them.



3. Party banner

Beaver Bert has a long strip of coloured paper for a party.
The strip has three different colours (yellow, red, blue) in a regularly repeating pattern.
Bert's friend, James, has cut out a section of the paper, as shown in the diagram below.

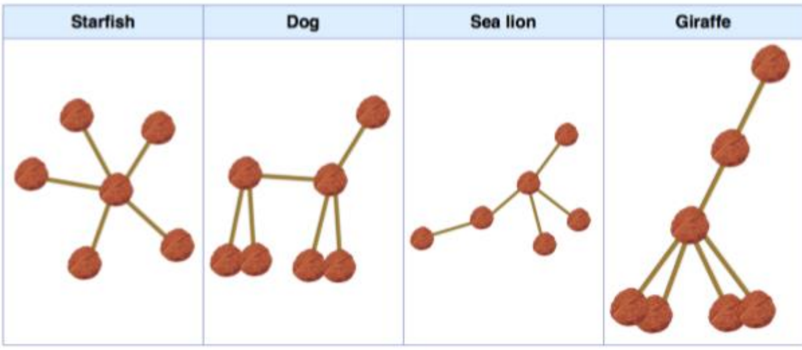
James says that he will give back the missing piece of paper if Bert can correctly guess the size of the piece cut out.

Question:
How many coloured squares does the missing piece of paper have?

31 32 33 or 34

4. Walnut animals

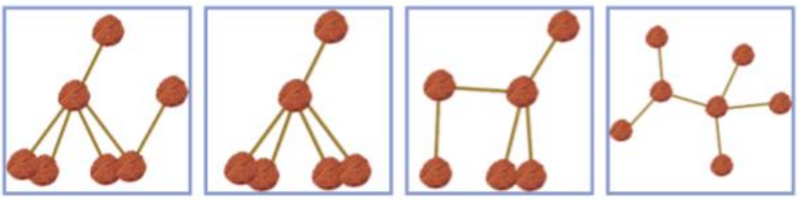
Gerald was playing in the woods. He used nuts and sticks to create four nice animals.



His sister managed to bend the animals around without removing any of the sticks.

Gerald was very upset because he really loved the figure of a dog.

Question:
Which of the following figures can be bent back to make the figure of the dog again?



5. Beaver Code

Barbara has been given two stamps.

With one she can produce a little flower, with the other a little sun.

Being a clever girl, she thinks of a way to write her own name by using the code below:

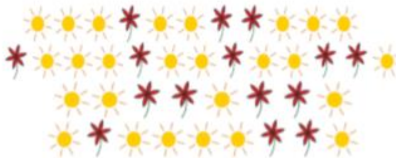
Letter	B	A	R	E	Y
Code					

So Barbara" becomes:

She then writes the names of her friends. Unfortunately they all got mixed up.

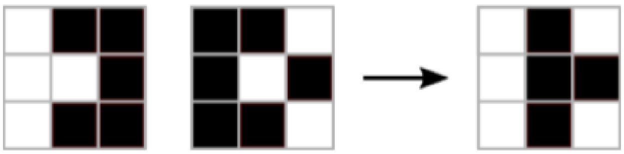
Question:
Drag the sun-flower-codes to the names of her four friends.

Abby	
Arya	
Barry	
Ray	



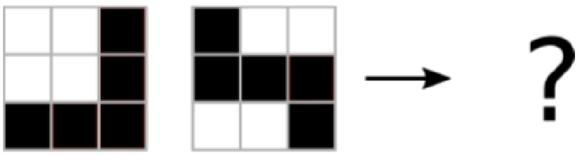
6. Paint it black

Combining Card A and Card B, you get Card C:

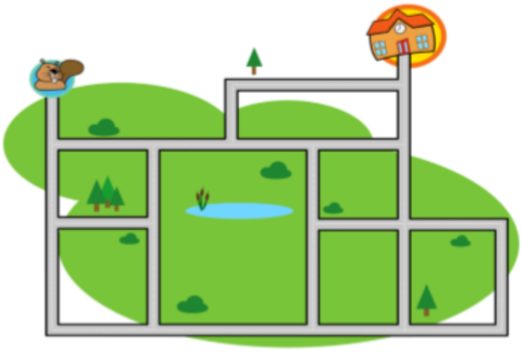


Card A Card B Card C

Question:
How many black cells will Card F have after combining Card D and Card E?



7. Car trip



A self-driving car needs to take a student to school.

The car is programmed so that it only use these 3 instructions:

Left: turn 90° left

Right: turn 90° right

Forward: go forward until you cannot go forward anymore

Question:

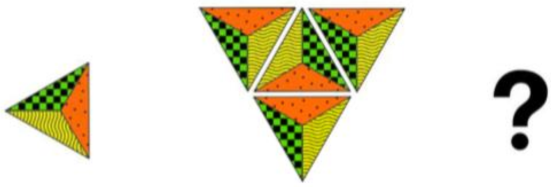
Write a set of instructions (a program) that will get the beaver to his school. You can do this by dragging the three instruction blocks next to the car.

8. Triangles

A beaver wants to create a mosaic with identical, triangle-shaped tiles.

He starts with one tile. He rotates it 90 degrees clockwise and then adds tiles on each side of the triangle-shaped tile, as shown in the picture below.

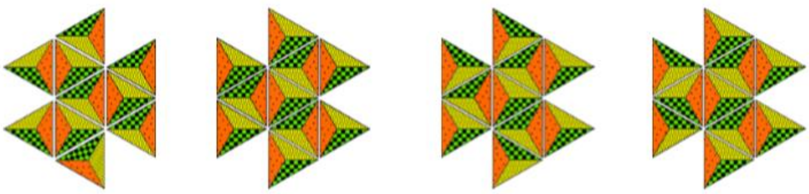
Then he rotates the whole shape 90 degrees clockwise again and adds tiles to the sides as before.



Step = 1 Step = 2 Step = 3

Question:

What will be the final shape of the triangles after step 3?



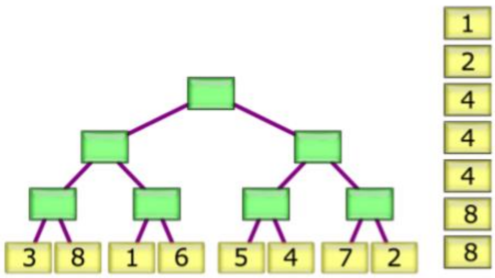
a b c d

9. Beaver Tournament

Beaver Krešo watched a tournament of races and recorded the winners of each stage on the board below.
The runners wore the same numbers, from 1 to 8, throughout the tournament.
Krešo used numbered cards to represent each runner.

When the tournament was over his younger brother Tomo mixed up all the cards, except those from the first stage of the tournament.

Task:
Can you put the labels in the correct positions ?



10. Magic potions

Betaro Beaver has discovered five new magic potions:

one makes ears longer
another makes teeth longer
another makes whiskers curly
another turns the nose white
the last one turns eyes white.

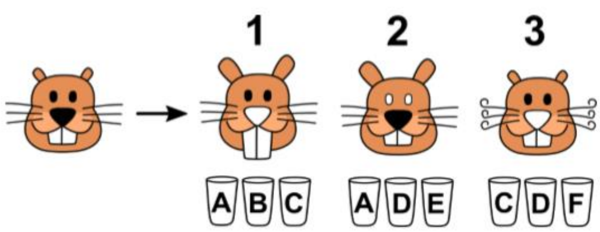
Betaro put each magic potion into a separate beaker. He put pure water into another beaker, so there are six beakers in total. The beakers are labeled A to F. The problem is, he forgot to record which beaker contains which magic potion!

To find out which potion is in each beaker, Betaro set up the following experiments:

Expt 1: A beaver drinks from beakers A, B and C together - the effects are shown in Figure 1.

Expt 2: A beaver drinks from beakers A, D and E together - the effects are shown in Figure 2.

Expt 3: A beaver drinks from beakers C, D and F together - the effects are shown in Figure 3.

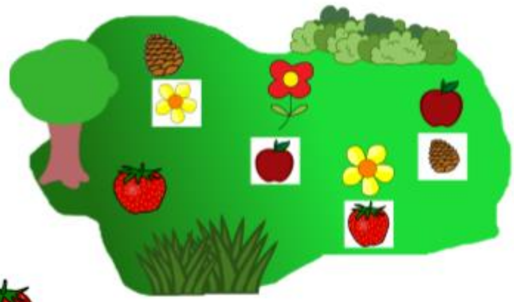


Question:
Which beaker contains pure water?

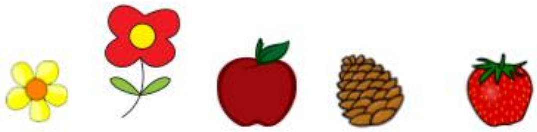
11. Secret recipe

Eszter has asked István to cook a special cake made of five ingredients.
She has put labels next to the ingredients in the garden. One ingredient has no label.
The labels tell István in what order the ingredients must be added.

The garden looks like this:



Question:
Which ingredient should be added first?



12. Quick beaver

The beavers want to encode numbers. They developed the Quick-Beaver-Code (QB-Code).

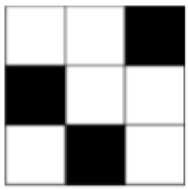
This is a code consisting of squares. Every square has a certain value.
The squares are filled line by line from the bottom to the top and from right to left.

The value of the bottom right square is 1. The other squares have double the value of the square before them.

Example:

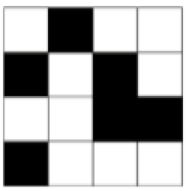
Here is a 3x3 QB-Code. The beavers have encoded a number by darkening some squares.

The number encoded is the sum of the values of the dark squares,
so the number encoded in this QB-Code is 2 + 32 + 64 = 98.

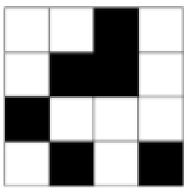


Question:

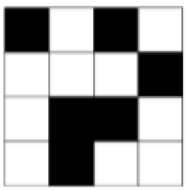
Of the following 4x4 QB-Codes, which one encodes the highest number?



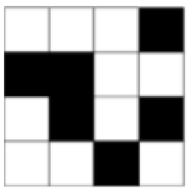
A



B



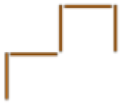
C



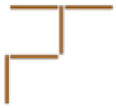
D

13. Five sticks

Adam has five sticks. He puts them on the table and creates this shape:



Nola comes to the table. She takes one stick and puts it in a different place:



Then Bob comes to the table, he also takes one stick and puts it in a different place.

Question:

Which shape is Bob not able to make?

A B C D

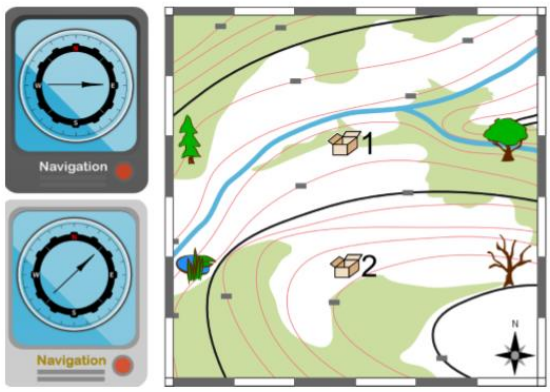
14. Geocaching

Two friends, Anna and Bob, are searching for treasure.

They have a smartphone app that shows them the direction to the treasure they are looking for.

The two boxes on the map show where the treasure is.

Anna is searching for box 1. Bob is looking for box 2.



Anna and Bob are standing in the same place. The picture shows the map and a screenshot of the smartphones.

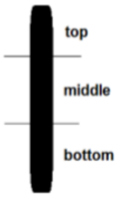
Question:

Where are Anna and Bob standing?

15. Kix code

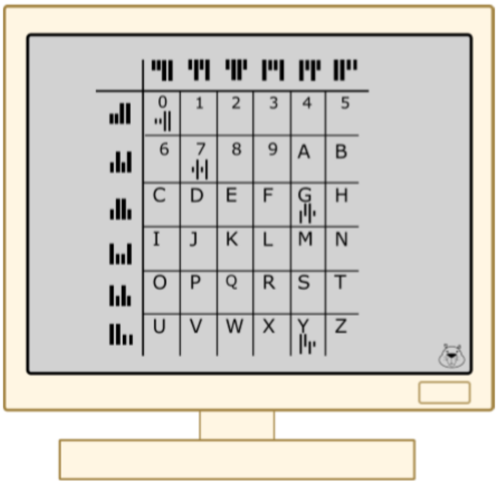
The Bebras Post Office uses postal codes that contain four characters.
To make the postal codes readable by machines, they convert the postal codes into Kix codes.

In a Kix code, each character is represented by 4 vertical bars.



A code has 2 sections: upper and lower.
The upper section contains only the middle and the top bars, while the lower section contains only the middle and the bottom bars.

This table shows the codes for several characters:



Example:
The Kix code for "G7Y0" is

Question:
Another postal code has this Kix code.

What is the postal code?

POST-TEST (TURKISH)

1. Dondurma

Bir dondurmacıda, dondurma topları müşterilerin istediği sırayla külaha koyulmaktadır.

Soru:

Yandaki şekilde bulunan dondurma külahını alan müşteri nasıl sipariş vermiştir?

A) çikolatalı, kivili ve çilekli

B) çikolatalı, çilekli ve kivili

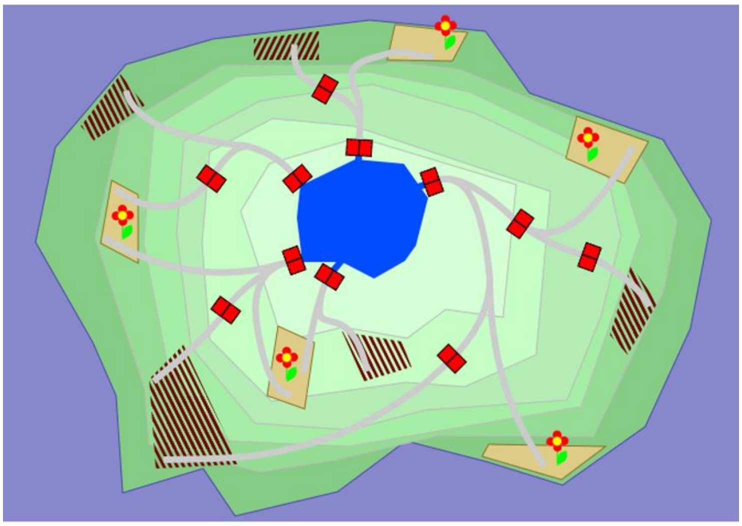
C) çilekli, kivili ve çikolatalı

D) çilekli, çikolatalı ve kivili

---

2. Baraj

Soru: Barajın etrafındaki tarlaların bazılarına çiçek, bazılarına ise tohum ekilmiş. Çiçek dikilen tarlaların sulanması, tohum ekilen tarlaların da kuru kalması gerekli. Suyun yalnızca çiçek olan tarlalara ulaşması için **en az kaç kapak açılmalı?** Yuvarlak içine alarak gösterin.



---

3. Süsleme Kağıdı

Kunduz Mert'in renkli şeritlerden oluşmuş bir süsleme kağıdı var. Bu kağıt üç farklı renkten (sarı, kırmızı, mavi) oluşuyor ve sürekli tekrar eden desenleri var. Mert'in arkadaşı Jale bu kağıdın aşağıda gösterildiği gibi bir bölümünü kesmiş.

Jale Mert'e diyor ki: "Eğer kestiğim parçanın uzunluğunu doğru tahmin edebilirsen, kestiğim parçayı sana geri vereceğim."

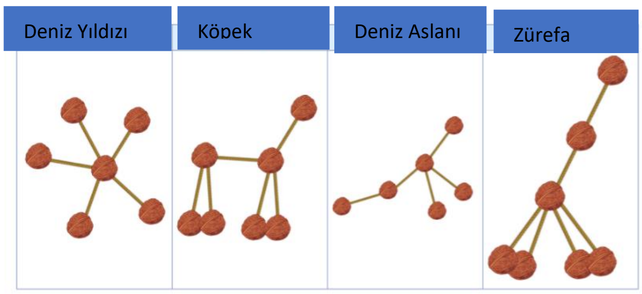
Soru: Kesilen parçanın uzunluğu hakkında senin tahminin aşağıdakilerden hangisi?

A) 31 B) 32 C) 33 D) 34

---

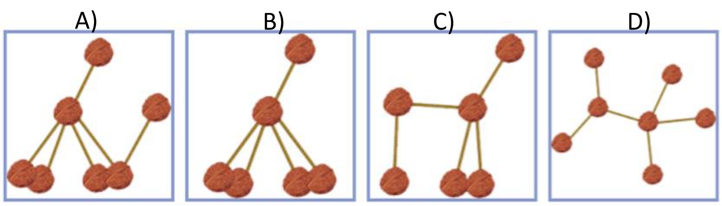
4. Çubuk Fındık Oyunu

Görkem bahçede oynuyordu. Fındıkları ve çubukları dört tane sevimli hayvan figürü yapmak için kullanmış.



Görkemin kardeşi hiçbir çubuğu çıkarmadan hayvan figürlerini bozdu. Görkem çok üzgündü. Çünkü köpek figürünü çok sevmişti.

Soru: Çubukların yerini değiştirmeden sadece bükerek tekrar köpek figürü aşağıdaki seçeneklerden hangisi ile yapılır?



---

5. Baskı

Berra’nın iki tane damgası var. Bir tanesi küçük bir çiçek, bir tanesi de küçük bir güneş basıyor. Akıllı bir kız olan Berra aşağıdaki kodları kullanarak kendi adını yazıyor.

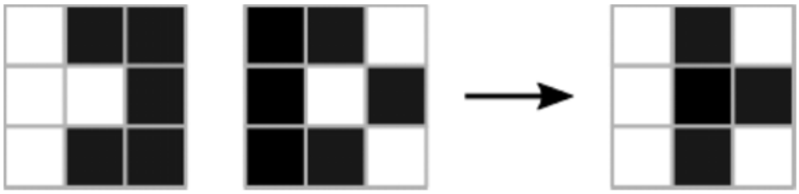
Harfler	B	A	R	E	Y
Kod					

BERRA kodları kullanarak ismini aşağıdaki gibi yazmıştır:

Soru: ARYA’yı çiçek ve yıldız kodlarını kullanarak yaz.

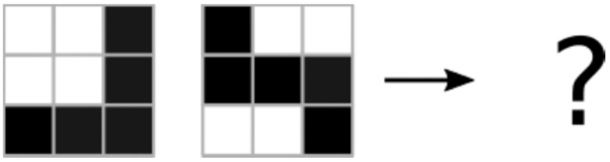
6. Gizemli Kareler

Aşağıdaki A ve B kartları belirli bir kurala göre işlem gördüğünde C kartı elde edilmektedir.



A B C

Soru: D ve E kartları, aynı kurala göre işlem gördüğünde elde edilecek kartta kaç adet siyah hücre bulunacaktır? Çizerek gösterin.

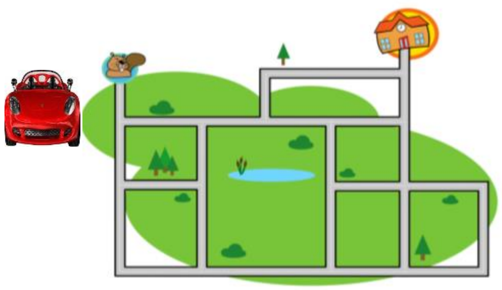


D E

A) 1 B) 2 C) 3 D) 4

---

7. Araba Yolculuğu



Sürücüsüz çalışan bir arabanın bir öğrenciyi okuluna götürmesi gerekiyor. Bu araç sadece aşağıdaki üç yönergeye uygun hareket etmek için programlanmış.

Sol: 90° sola dön

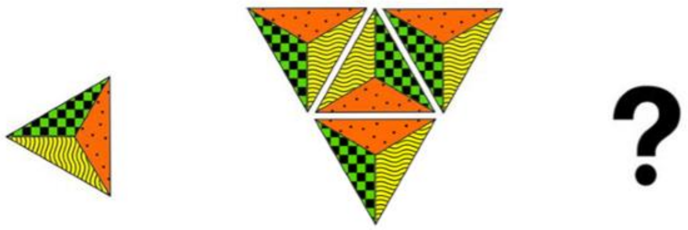
Sağ: 90° sağa dön

İleri: Yol bitene kadar ilerle

Soru: Kunduzu okula ulaştırmak için aracın takip etmesi gereken yolu şekil üzerinde çiz ve yönergeleri aşağıya yaz.

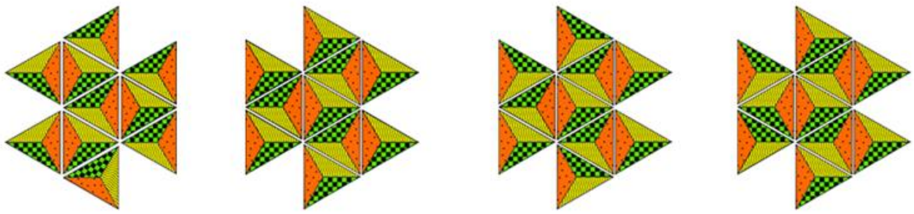
8. Üçgen Şekilleri

Bir kunduz aynı, üçgen şekilli çiniler ile bir mozaik oluşturmak istiyor. Bir taşla başlıyor. Saat yönünde 90 derece döndürür ve aşağıdaki resimde gösterildiği gibi üçgen şeklindeki döşemenin her bir tarafına döşemeler ekler. Sonra bütün şekli saat yönünde tekrar 90 derece döndürür ve daha önce olduğu gibi yanlara fayans ekler.



Adım 1 Adım 2 Adım 3

Soru: 3. adımdan sonra üçgenlerin son şekli ne olacak?

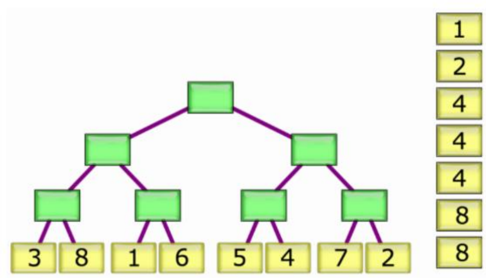


a b c d

9. Kunduz Turnuvası

Kunduz Kamil bir koşu turnuvasına izleyici olarak katılıyor. Turnuva boyunca her aşamadaki kazananları bir tabloya kaydediyor. Yarışmadaki koşucuların turnuva boyunca giydikleri forma numaraları değişmiyor. Kamil koşucuları temsil etmek için 1'den 8'e kadar numaralandırdığı kartları kullanıyor. Turnuva sonunda Kamil'in küçük kardeşi Taha ilk aşamadaki yarışma sonuçları hariç diğer aşamalardaki tüm kartları karıştırıyor.

Soru: Aşağıdaki boş kutulara yandaki kartlardan hangileri gelmelidir. Üzerlerine yaz.

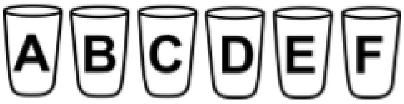


10. Sihirli Kaplar

Bilge Kunduz yeni büyü iksirleri keşfetti:

- Bunlardan biri kulakları daha uzun,
- diğeri dişleri daha uzun,
- diğeri bıyıkları kıvrık,
- diğeri burnu beyaz,
- diğeri gözleri beyaz yapar

Bilge Kunduz'un elinde altı kap var. Bunların 5'inde farklı iksirler, diğerinde ise saf su vardır. Bilge Kunduz her bir kaba A'dan F'ye kadar bir etiket vermiş. Fakat sonradan, hangi kapta hangi iksirin olduğunu unutmuş.

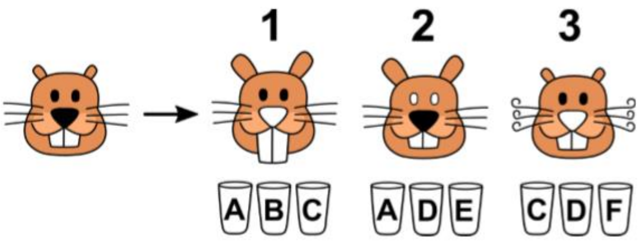


Her bir kapta hangi iksirin olduğunu bulmak için ise, aşağıdaki deneyleri hazırlıyor: Deneylerin sonuçları aşağıdaki şekillerde gösteriliyor.

Deney 1: A, B ve C bardaklarından bir kunduz içiyor - etkileri Şekil 1'de gösteriliyor.

Deney 2: A, D ve E bardaklarından bir kunduz içiyor - etkiler Şekil 2'de gösterilmektedir.

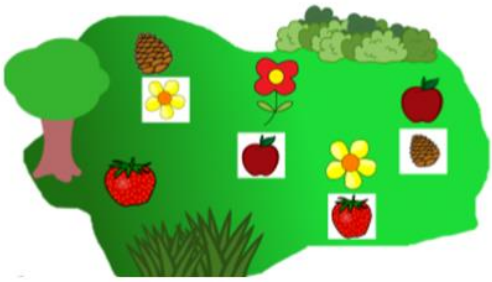
Deney 3: C, D ve F bardaklarından bir kunduz içiyor - etkiler Şekil 3'te gösterilmektedir.



Soru: Saf suyun hangi kapta olduğunu açıklayın ve nasıl bulduğunuzu açıklayın.

11. Özel Tarif

Esra İrem'den 5 malzemeden oluşan özel bir kek pişirmesini istiyor. Bunun için bahçedeki malzemelerin yanlarına bir sonraki eklenecek malzemeyi gösteren etiketler yerleştiriyor. Ancak, malzemelerden bir tanesinin etiketi eksik. Bahçenin görüntüsü aşağıdaki resimdeki gibidir.



Soru: İlk eklenecek malzeme aşağıdakilerden hangisidir?

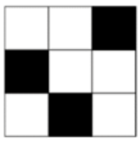
A. Çilek B. Kırmızı çiçek C. Sarı çiçek D. Elma

---

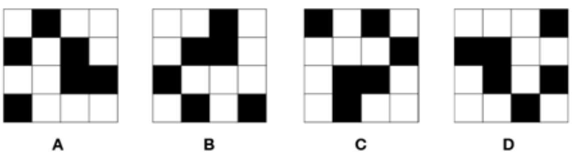
12. Hızlı Kunduz Kodu

Kunduzlar sayıları kodlamak istiyor. Bunun için Hızlı Kunduz kodu (HK) adını verdikleri, karelerden oluşan bir kod geliştirdiler. Bu kodda her karenin belirli bir değeri var. **Kareler alttan üste ve sağdan sola doğru sırayla dizilir**. Sağ alt karenin değeri 1'dir. Diğer kareler kendilerinden bir önceki karenin değerini ikiye katlar.

Örnek: 3x3 HK Kodu. Kunduzlar bazı kareleri siyah renkle boyayıp kodlamışlardır. Kodlanan sayı, siyah karelerin değerlerinin toplamıdır, bu nedenle bu HK Kodunda kodlanan sayı 2 + 32 + 64 = 98'dir.



Soru: Aşağıdaki 4x4 Hızlı Kunduz Kodlarından hangisi en yüksek sayıyı kodlar?



---

13. Beş Çubuk

Ahmet, Ali ve Emel birlikte bir oyun oynuyorlar.

Ahmet'in elinde beş ayrı çubuk var. Çubukları birleştirerek aşağıdaki şekli yapıyor.

Daha sonra Ali gelip Ahmet'in yaptığı şekildeki **sadece bir çubuğun** yerini değiştiriyor. Ve aşağıdaki şekli oluşturuyor.

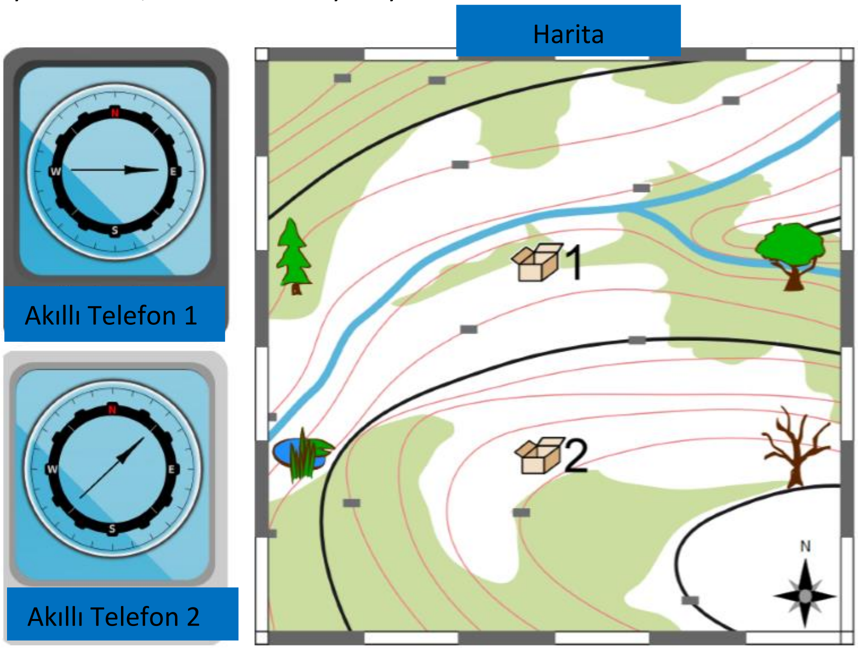
Daha sonra da Emel gelip Ali'nin yaptığı şekildeki **sadece bir çubuğun** yerini değiştirerek yeni bir şekil oluşturuyor.

Soru: Emel sadece bir çubuğun yerini değiştirerek aşağıdaki şekillerden hangisini yapamaz?

A B C D

14. Hazine Arama

İki arkadaş, Ayla ve Bora, hazine arıyorlar. Onlara aradıkları hazinenin yönünü gösteren bir akıllı telefon uygulaması var. Haritadaki iki kutu aradıkları hazinelerin nerede olduğunu gösteriyor. Ayla kutu 1'i , Bora ise kutu 2'yi arıyor.



Ayla ve Bora aynı yerde duruyorlar. Resimde harita ve akıllı telefonların ekran görüntüsü gösterilmektedir.

Soru: Ayla ve Bora nerede duruyor?

A) 

B) 

C) 

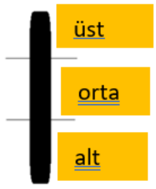
D) 

15. Posta Kutusu Kodlama

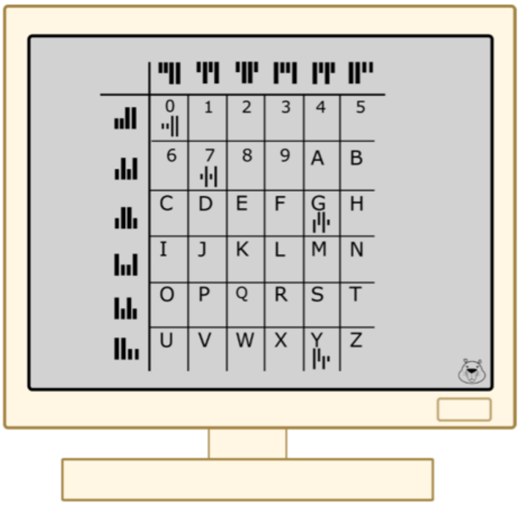
Bilge Kunduz Postası, dört karakter içeren posta kodlarını kullanır. Posta kodlarını makineler tarafından okunabilir hale getirmek için, posta kodlarını Kunduz kodlarına dönüştürürler. Bir Kunduz kodunda, her karakter 4 dikey çubukla gösterilir.

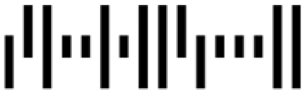
Bir kod 2 bölümden oluşur: üst ve alt.

Üst kısım sadece orta ve üst çubuklar içerirken, alt kısım sadece orta ve alt çubuklar içerir.

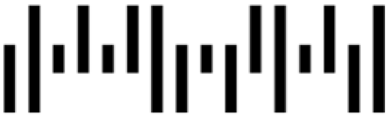


Bu tabloda birkaç karakterin kodları gösterilmektedir:



Örneğin "G7Y0" in Kunduz kodu şekildeki gibidir. 

Soru: Kunduz kodu aşağıda verilen şeklin posta kodunu nedir, yazın.



APPENDIX D

KORKMAZ AND COLLEAGUES' CT SCALE

Computational Thinking Scale for Middle School Students

Fact.	Items					
C1	I like the people who are sure of most of their decisions	1	2	3	4	5
C4	I have a belief that I can solve the problems possible to occur when I encounter with a new situation	1	2	3	4	5
C5	I trust my intuitions and feelings of "trueness" and "wrongness" when I approach the solution of a problem	1	2	3	4	5
C8	When I encounter with a problem, I stop before proceeding to another subject and think over that problem	1	2	3	4	5
A1	I can immediately establish the equity that will give the solution of a problem	1	2	3	4	5
A3	I think that I learn better the instructions made with the help of mathematical symbols and concepts	1	2	3	4	5
A4	I believe that I can easily catch the relation between the figures	1	2	3	4	5
A6	I can digitize a mathematical problem expressed verbally.	1	2	3	4	5
O1	I like experiencing cooperative learning together with my group friends.	1	2	3	4	5
O2	In the cooperative learning, I think that I attain/will attain more successful results because I am working in a group.	1	2	3	4	5
O3	I like solving problems related to group project together with my friends in cooperative learning.	1	2	3	4	5
O4	More ideas occur in cooperative learning.	1	2	3	4	5
T1	I am good at preparing regular plans regarding the solution of the complex problems.	1	2	3	4	5
T2	It is fun to try to solve the complex problems.	1	2	3	4	5
T3	I am willing to learn challenging things.	1	2	3	4	5
T5	I make use of a systematic method while comparing the options at my hand and while reaching a decision.	1	2	3	4	5
P1	I have problems in the demonstration of the solution of a problem in my mind.	1	2	3	4	5
P2	I have problems in the issue of where and how I should use the variables such as X and	1	2	3	4	5

KORKMAZ AND COLLEAGUES' CT SCALE (TURKISH)

Bilgisayarca Düşünme Ölçeği (Ortaokul Düzeyi İçin)

Sevgili Öğrenciler

Aşağıdaki maddeler bilgisayarca düşünme becerilerini ölçmeye dönük hazırlanmış ve bir araştırmada kullanılacaktır. Araştırma dışında başka hiçbir amaçla kullanılmayacaktır. Lütfen her bir maddeyi dikkatle okuyup, sizi yansıtma düzeyini en olumludan (5) en olumsuza (1) doğru puanlayınız.

Katılımınızdan dolayı şimdiden teşekkür ederiz.

C1	Kararlarının çoğundan emin olan insanları severim	1	2	3	4	5
C4	Yeni bir durumla karşılaştığımda ortaya çıkabilecek sorunları çözebileceğime inancım vardır.	1	2	3	4	5
C5	Bir sorunumu çözmek üzere plan yaparken o planı yürütebileceğime güvenirim.	1	2	3	4	5
C8	Bir sorunla karşılaştığımda, başka konuya geçmeden önce durur ve o sorun üzerinde düşünürüm.	1	2	3	4	5
A1	Bir problemin çözümünü verecek denklemi hemen kurabilirim	1	2	3	4	5
A3	Matematiksel sembol ve kavramlar yardımıyla yapılan anlatımları daha kolay öğrendiğimi düşünürüm	1	2	3	4	5
A4	Sayılar arasındaki ilişkileri kolaylıkla yakalayabildiğime inanırım	1	2	3	4	5
A6	Sözel olarak ifade edilen bir matematik problemini sayısallaştırabilirim.	1	2	3	4	5
O1	Grup arkadaşlarımla birlikte işbirlikli öğrenme deneyimleri yaşamaktan hoşlanırım.	1	2	3	4	5
O2	İşbirlikli öğrenmede, grupla çalıştığım için daha başarılı sonuçlar elde ettiğimi/edeceğimi düşünüyorum.	1	2	3	4	5
O3	İşbirlikli öğrenmede grup arkadaşlarımla birlikte grup projesi ile ilgili problemleri çözmekten hoşlanırım.	1	2	3	4	5
O4	İşbirlikli öğrenmede daha çok fikir ortaya çıkıyor.	1	2	3	4	5
T1	Karmaşık problemlerin çözümüne yönelik düzenli planlar geliştirmede iyiyimdir.	1	2	3	4	5
T2	Karmaşık problemleri çözmeye çalışmak eğlencelidir.	1	2	3	4	5
T3	Zorlayıcı şeyler öğrenmeye istekliyimdir.	1	2	3	4	5
T5	Elimdeki seçenekleri karşılaştırırken ve karar verirken kullandığım sistematik bir yöntem vardır.	1	2	3	4	5
P1	Problemin çözümünü zihnimde canlandırma konusunda sıkıntı yaşarım.	1	2	3	4	5

P2	Problem çözümünde X, Y gibi değişkenleri nerede ve nasıl kullanmam gerektiği konusunda sıkıntı yaşarım.	1	2	3	4	5
P3	Tasarladığım çözüm yollarını sırasıyla aşamalı bir şekilde uygulayamam.	1	2	3	4	5
P4	Bir soruna yönelik olası çözüm yollarını düşünürken çok fazla seçenek üretemem.	1	2	3	4	5
P5	İşbirlikli öğrenme ortamında kendi düşüncelerimi geliştiremem.	1	2	3	4	5
P6	İşbirlikli öğrenme grup arkadaşlarıma bir şeyler öğretmeye çalışmak beni yoruyor.	1	2	3	4	5

APPENDIX E

APPROVAL FORM

T.C.
BOĞAZİÇİ ÜNİVERSİTESİ
Sosyal ve Beşeri Bilimler Yüksek Lisans ve Doktora Tezleri Etik İnceleme Komisyonu

Sayı: 2019-16-A

7 Mart 2019

Havva Delal
Bilgisayar ve Öğretim Teknolojileri Eğitimi

Sayın Araştırmacı,

"Bilgisayarsız bilgisayar bilimi etkinliklerinin öğrencilerin bilgi işlemsel düşünme becerilerine etkisi" başlıklı projeniz ile ilgili olarak yaptığınız SBB-EAK 2018/35 sayılı başvuru komisyonumuz tarafından 7 Mart 2019 tarihli toplantıda incelenmiş ve uygun bulunmuştur.

Dr. Öğr. Üyesi İnci Ayhan

Prof. Dr. Feyza Çorapçı

Doç. Dr. Mehmet Yiğit Gürdal

Doç. Dr. Ebru Kaya

Dr. Öğr. Üyesi Şebnem Yalçın

REFERENCES

Aharoni, D. (2000). Cogito, Ergo sum! Cognitive processes of students dealing with data structures. *ACM SIGCSE Bulletin*, *32*(1), 26-30.

Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. *Robotics and Autonomous Systems*, *75*, 661-670. doi: 10.1016/j.robot.2015.10.008

Barendsen, E., Mannila, L., Demo, B., Grgurina, N., Izu, C., Mirolo, C., & Stupurienė, G. (2015). Concepts in K-9 computer science education. In A. Korhonen (Eds.), *Proceedings of the 2015 ITiCSE on Working Group Reports* (pp. 85-116). Vilnius, Lithuania: Vilnius University. Retrieved from https://dl.acm.org/citation.cfm?id=2858800

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community? *ACM Inroads, 2*(1), 48- 54.

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2008). Computer science without computers: new outreach methods from old tricks. In S. Mann, & M. Lopez (Eds.), *Proceedings of the 21st Annual Conference of the National Advisory Committee on Computing Qualifications (NACCQ 2008)* (pp. 127-133). Auckland, New Zealand: NACCQ. Retrieved from http://www.citrenz.ac.nz/conferences/2008/127.pdf

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology*, *13* (1), (pp. 20-29).

Bell, T., Andreae, P., & Lambert, L. (2010). Computer science in New Zealand high schools. In T. Clear, & J. Hamer (Eds.), *Proceedings of the Twelfth Australasian Conference on Computing Education* (pp. 15-22). Brisbane, Australia: ACE. Retrieved from https://dl.acm.org/citation.cfm?id=1862223

Bell, T., Witten, I., & Fellows, M. (1998). Computer science unplugged. Retrieved from www.csunplugged.org

Bernard, H. R. (2002). *Research methods in anthropology: Qualitative and quantitative approaches* (3rd ed.). Walnut Creek, CA: Alta Mira Press.

Bilge Kunduz Report 2017 (n.d.) Retrieved June 1, 2019, from Web Site: http://www.bilgekunduz.org/wp-content/uploads/2015/10/bilgekunduz_2017_bilgi_grafigi.pdf

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In A. F., Ball, & C. A., Tyson (Eds.), *In Proceedings of the 2012 Annual Meeting of the American Educational Research Association* (pp. 1-25). Vancouver, Canada: AERA. Retrieved from http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

Brown, N. C. C., Kölling, M., Crick, T., Peyton Jones, S., Humphreys, S., & Sentance, S. (2013). Bringing computer science back into schools. *Proceeding of the 44th ACM Technical Symposium on Computer Science Education,* 269-274.

Burke, Q. (2012). The markings of a new pencil: Introducing programming-as-writing in the middle school classroom. *Journal of Media Literacy Education*, *4*(2), 121-135.

Calcagni, A., Lonati, V., Malchiodi, D., Monga, M., & Morpurgo, A. (2017). Promoting computational thinking skills: Would you use this Bebras task? In V. Dagiene & A. Hellas (Eds.), *In informatics in schools: Focus on learning programming*, (pp. 102-113), Helsinki, Finland: Springer.

Carlisle, M. C., Wilson, T. A., Humphries, J. W., & Hadfield, S. M. (2005). RAPTOR: a visual programming environment for teaching algorithmic problem solving. *Proceedings of the 36th SIGCSE technical symposium on Computer science education*, *37*(1), 176-180.

Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Lawrence Earlbaum Associates.

Computing at School & NAACE (2014). *Computing in the national curriculum: A guide for secondary teachers*. Retrieved from https://goo.gl/qFz5NA

Cooper, S., Dann, W., & Pausch, R. (2000). Developing algorithmic thinking with Alice. T*he proceedings of ISECON, 2000*(5)*,* 506-539.

Cortina, T. J. (2015). Reaching a broader population of students through "unplugged" activities. *Communications of the ACM, 58*(3), 25-27.

Creswell, J.W. (2003). *Research design – qualitative, quantitative and mixed method approaches* (2nd ed.). Thousand Oaks, CA: Sage Publications.

CSTA. (2011). *Computer science standards* (SBN: 978-1-4503-0881-6). Retrieved from https://goo.gl/im8r5l

Cuny, J., Snyder, L., & Wing, J.M. (2010). *Demystifying computational thinking for noncomputer scientists*. Unpublished manuscript in progress, retrieved from http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf

Curzon, P., Black, J., Meagher, L. R., & McOwan, P. W. (2009). cs4fn.org: Enthusing Students about Computer Science. *Proceedings of Informatics Education Europe IV*, 73-80.

Curzon, P., & McOwan, P. W. (2008). Engaging with computer science through magic shows. *Proceedings of the 13th Annual Conference on Innovation and Technology in Computer Science Education*, *40*(3), 179-183.

Curzon, P., McOwan, P. W., Plant, N., & Meagher, L. R. (2014). Introducing teachers to computational thinking using unplugged storytelling. In J. Gal-Ezer, S. Sentance, & J. Vahrenhold (Eds.), *Proceedings of the 9th Workshop in Primary and Secondary Computing Education* (pp. 89-92). London, United Kingdom: ACM. Retrieved from https://dl.acm.org/citation.cfm?id=2670767

Çetin, İ., & Uçar Toluk, Z. (2018). Bilgi işlemsel düşünme tanımı ve kapsamı. In Y. Gülbahar (Eds.), *Bilgi işlemsel düşünmeden programlaya kitabı* (pp. 41-74). Ankara, Turkey: PEGEM Akademi.

Gülbahar, Y., Kalelioğlu, F., & Doğan, D. (2016). *Bilge Kunduz Uluslarası Enformatik ve Bilg İşlemsel Düşünme Etkinliği: 2015 Yılı Uygulama Raporu*. Retrieved from http://www.bilgekunduz.org/wp-content/uploads/2016/01/bilgekunduz-rapor-2015.pdf

Dagiene, V., & Futschek, G. (2008). Bebras international contest on informatics and computer literacy: Criteria for good tasks. In Mittermeir, R.T., & Syslo, M.M. (Eds.), *Informatics education – supporting computational thinking*, (pp. 102-113), Torun, Poland: Springer.

Dagiene, V., Mannila, L., Poranen, T., Rolandsson, L., & Stupuriene, G. (2014). Reasoning on children's cognitive skills in an informatics contest: findings and discoveries from Finland, Lithuania, and Sweden. In Y. Gülbahar & E. Karataş (Eds.), *Informatics in education*, (pp. 25 - 44), Istanbul, Turkey: Springer.

Dagiene, V., Sentence, S., & Stupuriene, G. (2017). Developing a Two-Dimensional Categorization System for Educational Tasks in Informatics. *Informatica, 28* (1), 23-24.

Dagiene, V., & Stupuriene, G. (2016). Bebras - a Sustainable Community Building Model for the Concept based Learning of Informatics and Computational Thinking. *Journal of Eastern and Central Europe, 15*(1), 25-44.

Dagiene, V., Stupuriene, G., Vinikiene, L., & Zakauskas, R. (2017). Introduction to Bebras challenge management: Overview and analyses of developed systems. In V. Dagiene & A. Hellas (Eds.), *Informatics in schools: Focus on learning programming,* (pp. 232-243), Helsinki, Finland: Springer.

Denning, P. J., (2007). An Interview with Peter Denning on the great principles of computing. *Ubiquity*, 2007, 1-1. Received from https://core.ac.uk/download/pdf/81223217.pdf

Futschek, G. (2006). Algorithmic thinking: the key for understanding computer science. In R. T. Mittermeir (Eds.), *Informatics education – the bridge between using and understanding computers,* (pp. 7-11), Vilnius, Lithuania: Springer.

Howarth, L., Millican,P., Roffey, C., & Sentance, S. (2014). UK Bebras computational thinking challenge. Retrieved from http://www.bebras.uk/uploads/2/1/8/6/21861082/juniorukbebras2014-answers_1.pdf

Hubwieser, P. (2012). Computer science education in secondary schools -- The introduction of a new compulsory subject. *ACM Transactions on Computing Education, 12*(4), 1–41.

Hubwieser, P., Hubwieser, E., & Graswald, D. (2016). How to attract the girls: gender-specific performance and motivation in the Bebras challenge. In A. Brodnik, & F. Tort (Eds.), *Informatics in Schools: Improvement of Informatics Knowledge and pPerception,* (pp. 40-52), Münster, Germany: Springer. Retrieved from https://link.springer.com/chapter/10.1007/978-3-319-46747-4_4

Hubwieser, P., & Mühling, A. (2015). Investigating the psychometric structure of Bebras contest: Towards measuring computational thinking skills. In A. Pears, & C.C. Wu (Eds.), *2015 International Conference on Learning and Teaching in Computing and Engineering* (pp. 62-69). Taipei, Taiwan: IEEE. Retrieved from https://ieeexplore.ieee.org/abstract/document/7126233

ISTE & CSTA (2011). Operational definition of computational thinking for K-12 education. Retrieved from http://www.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf

Izu, C., Mirolo, C., Settle, A., Mannila, L., & Stupurienė, G. (2017). Exploring Bebras tasks content and performance: A multinational study. *Informatics in Education*, *16* (1), 39-59.

Kalaš, I., Tomcsányiová, M. (2009). Students' attitude to programming in modern informatics. *Informi¿tica na Educa-ção: Teoria & Prática, Porto Alegre*, *12*(1), 12-135.

Kalelioğlu, F. (2018). Bilgisayarsız Bilgisayar Bilimi (B^3) Öğretimi. In Y. Gülbahar (Eds.), *Bilgi İşlemsel Düşünmeden Programlaya Kitabı* (pp. 183-204). Ankara, Turkey: PEGEM Akademi.

Kalelioglu, F., Gülbahar, Y., & Kukul, V. (2016). A Framework for Computational Thinking Based on a Systematic Research Review. *Baltic Journal of Modern Computing*, *4*(3), 583.

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming. *ACM Computing Surveys, 37*(2), 83–137.

Kert, S., B. (2018). Bilgisayar bilimi eğitimine giriş. In Y. Gülbahar (Eds.), *Bilgi işlemsel düşünmeden programlaya kitabı* (pp. 1-20). Ankara, Turkey: PEGEM Akademi.

Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2016). Bilgisayarca düşünme beceri düzeyleri ölçeğinin (BDBD) ortaokul düzeyine uyarlanması. *Gazi Eğitim Bilimleri Dergisi*, *1*(2), 143-162.

Korkmaz, Ö., Çakır, R., Özden, M., Oluk, A. & Sarıoğlu, S. (2015). Bireylerin bilgisayarca düşünme becerilerinin farklı değişkenler açısından incelenmesi. *Ondokuz Mayıs Üniversitesi Eğitim Fakültesi Dergisi*, *34*(2), 68-87.

Lee, Y.-J. (2010). Developing computer programming concepts and skills via technology-enriched language-art projects: A case study. *Journal of Educational Multimedia and Hypermedia*, *19*(3), 307-326.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads*, *2*(1), 32-37.

Liskov, B., & Guttag, J., (2000). *Program development in java: Abstraction, specification, and object-oriented design* (1st ed.). Boston, MA: Addison-Wesley.

Lu, J.J., & Fletcher, G.H. (2009). Thinking about computational thinking. *ACM SIGCSE Bulletin*, *41*(1), 260-264.

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior, 41*, 51-61. doi:10.1016/j.chb.2014.09.012

Mannila, L., Dagiene, V., Demo, B., Grgurina, N., Mirolo, C., Rolandsson, L., & Settle, A. (2014). Computational thinking in K-9 education. In T. Clear, & A. Pears (Eds.). *Proceedings of the Working Group Reports of the 2014 on Innovation & Technology in Computer Science Education Conference* (pp. 1-29). Uppsala, Sweden: ACM. Retrieved from https://dl.acm.org/citation.cfm?id=2713610

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2013). Learning computer science concepts with Scratch. *Computer Science Education*, *23*(3), 239-264.

Milli Eğitim Bakanlığı (2018). Bilişim teknolojileri ve yazılım dersi öğretim programı (ortaokul 5. ve 6. sınıflar). Retrieved from http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1l%C4%B1m%205-6.%20S%C4%B1n%C4%B1flar.pdf

Moreno-León, J., & Robles, G. (2015). Dr. Scratch: A web tool to automatically evaluate scratch projects. In J. Gal-Ezer, S. Sentance, & J. Vahrenhold (Eds.), *Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 132-133). London, United Kingdom: ACM. Retrieved from https://www.researchgate.net/profile/Jesus_Moreno-Leon/publication/284181364_Dr_Scratch_a_Web_Tool_to_Automatically_Evaluate_Scratch_Projects/links/564eccb508aefe619b0ff212.pdf

National Research Council, (2010). Report of a workshop on the scope and nature of computational thinking. Retrieved from http://www.nap.edu/catalog.php?record_id=12840.

Nishida, T., Kanemune, S., Idosaka, Y., Namiki, M., Bell, T., & Kuno, Y. (2009). A CS unplugged design pattern. *ACM SIGCSE Bulletin*, *41*(1), 231.

Opmanis, M., Dagiene, V., & Truu, A. (2006). Task types at "Beaver" contests. In V. Dagien, & R. Mittermeir (Eds.), *Proceedings of the 2nd Int. Conference Informatics in Secondary Schools: Evolution and Perspectives* (pp. 509-519). Vilnius, Lithuania: ACM. Retrieved from https://pdfs.semanticscholar.org/d8ea/9a95c392c9300eb97a7cf46707ebd715a3a5.pdf

Papert, S. (1996). An exploration in the space of mathematics educations. *International Journal of Computers for Mathematical Learning, 1*(1), 95-123.

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

Riley, D. D., & Hunt, K. A. (2014). *Computational thinking for the modern problem solver* (1st ed.). Boca Raton, FL: CRC Press.

Rodriguez, B., (2015). *Assessing computational thinking in computer science unplugged activities* (Unpublished master dissertation). Colorado School of Mines, Golden, America.

Rodriguez, B., Rader, C., & Camp, T. (2016). Using student performance to assess CS unplugged activities in a classroom environment. In E. Cuadros-Vargas, J. Carter, & Y. Tupac (Eds.), *Proceedings of the 2016 ACM Conference on Innovation and Technology in Computer Science Education* (pp.95-100).

Arequipa, Peru: ACM. Retrieved from https://dl.acm.org/citation.cfm?id=2899465

Sayın, Z. (2018). Bilgisayar bilimi eğitimi kapsamı. In Y. Gülbahar (Eds.), *Bilgi işlemsel düşünmeden programlaya kitabı* (pp. 133-153). Ankara, Turkey: PEGEM Akademi.

Seiter, L., & Foreman, B. (2013). Modeling the learning progressions of computational thinking of primary grade students. In B. Simon, A. Clear, & Q. Cutts (Eds.), *Proceedings of the ninth annual international ACM conference on International computing education research* (pp. 59-66). San Diego, San California: USA. Retrieved from https://dl.acm.org/citation.cfm?id=2493403

Sekaran, U. (1992). Research methods for business: A skill-building approach. In New York: John Wiley & Sons.

Selby, C., & Woollard, J. (2013). Computational thinking: the developing definition. Retrieved from http://eprints.soton.ac.uk/356481.

Taub, R., Ben-Ari, M., & Armoni, M. (2009). The effect of CS unplugged on middle-school students' views of CS. *Proceedings of the 14th Annual ACM SIGCSE Conference on Innovation and Technology in Computer Science Education, 41*(3), 99-103.

Tebliğler Dergisi, (2013). 2669 Sayılı Tebliğler Dergisi, Retrieved from https://goo.gl/35FonN.

Tharenou, P., Donohue, R., & Cooper, B. (2007). *Management Research Methods* (1st ed.), Cambridge: Cambridge University Press.

Thies, R., & Vahrenhold, J. (2013). On plugging "unplugged" into CS classes. In J. D. Dougherty, & K. Nagel (Eds.), *Proceeding of the 44th ACM Technical Symposium on Computer Science Education* (pp. 365-370). Denver, Colorado: USA. Retrieved from https://dl.acm.org/citation.cfm?id=2445303

UK Bebras Computational Thinking Challenge 2017 Answers (2017). Received from http://www.bebras.uk/uploads/2/1/8/6/21861082/uk-bebras-2017-answers.pdf

Wilson, A., Hainey, T., & Connolly, T. (2012). Evaluation of Computer Games Developed by Primary School Children to Gauge Understanding of Programming Concepts. In F. Felicia (Eds.), *Proceedings of the 6th European Conference on Games-Based Learning* (pp. 4-5). Cork, Ireland: Academic Publishing International Limited.

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

Wing, J. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717-3725.

Wohl, B., Porter, B., & Clinch, S. (2015). Teaching computer science to 5-7-year-olds: An initial study with scratch, cubelets and unplugged computing. In J. Gal-Ezer, S. Sentance, & J. Vahrenhold (Eds.), *Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 55-60). London, United Kingdom: ACM. Retrieved from https://dl.acm.org/citation.cfm?id=2818340