ASSESSING COMPUTATIONAL THINKING BEYOND PROGRAMMING: A TEST DEVELOPMENT STUDY FOR GRADES 5 AND 6

by

Beyzanur Yalvaç

B.S., Undergraduate Program in Mathematics Education, Boğaziçi University, 2022

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Mathematics and Science Education
Boğaziçi University
2025

ACKNOWLEDGEMENTS

First and foremost, I am sincerely thankful to my advisor, Assoc. Prof. Serkan Özel, for his continuous support, openness, approachability, and detailed feedback throughout this process. His friendly, guiding, and exemplary personality has strengthened my hope for the academic world.

I am deeply grateful to my mother, who has always supported me to reach this point with her guidance and compassion. Also, I would like to thank my father and brother, whose presence has been a constant source of reassurance, knowing that they would always be there if I needed them.

My heartfelt appreciation goes to my friends: Melike, for her invaluable support; Betül, Züleyha, and Belgin, for walking this path with me and making it easier to bear. I also thank Seher Bozdağlı, Fatih Delen, and Fikri Kaşifhüseyin for their help in data collection, and Dr. Gamze Kartal for her guidance in data analyses. I would also like to thank the students, the main actors of this study, and the jury members, Assist. Prof. Osman Akşit and Assoc. Prof. Selin Urhan for their kind participation.

I would also like to thank the TÜBİTAK 2210-A MSc/MA Scholarship Program and the TEKFEN Scientific Research Fund (Grant Number: 24D3TFV1) for supporting this study.

Most importantly, I am profoundly grateful to Allah (SWT) for granting me the grace, strength, perseverance, and patience to complete this thesis; without Her support, I would never have been able to complete this thesis.

Finally, I would like to extend my sincere thanks to all those whose names may not be mentioned here but who have contributed directly or indirectly to this work.

ABSTRACT

ASSESSING COMPUTATIONAL THINKING BEYOND PROGRAMMING: A TEST DEVELOPMENT STUDY FOR GRADES 5 AND 6

This study aimed to develop a valid and reliable assessment tool to measure Computational Thinking (CT) skills among 5th and 6th-grade students, based on a four-component theoretical framework: abstraction, algorithm, pattern recognition, and decomposition. An initial pool of 41 items was developed and subjected to expert review for content validity, resulting in a 26-item version. After a small-scale pilot study, the final version of the test was administered to 429 students. Exploratory Factor Analysis (EFA) and Confirmatory Factor Analysis (CFA) were conducted to investigate the construct validity. The reliability analysis revealed low internal consistency for each subscale (Cronbach's $\alpha = 0.30$ - 0.50), suggesting the need for more items per component. Additionally, Multidimensional Item Response Theory (MIRT) analysis under the 2-Parameter Logistic Model provided detailed information on item discrimination and difficulty. While the revised model showed acceptable fit, the low internal consistency across subscales limits the tool's current utility for diagnosing individual CT components. Nevertheless, the study contributes a promising prototype for holistic CT assessment, and the findings offer critical insights into the empirical challenges of translating theoretical CT constructs into psychometrically sound tools in unplugged contexts. These findings highlight the need for further refinement of both the theoretical model and item pool to support the development of valid and reliable instruments for CT assessments in educational settings.

ÖZET

PROGRAMLAMANIN ÖTESİNDE BİLGİ İŞLEMSEL DÜŞÜNMENİN DEĞERLENDİRİLMESİ: 5. VE 6. SINIFLAR İÇİN BİR TEST GELİŞTİRME ÇALIŞMASI

Bu çalışma, 5. ve 6. sınıf öğrencilerinin Bilgi İşlemsel Düşünme (BİD) becerilerini ölçmek için geçerli ve güvenilir bir değerlendirme aracı geliştirmeyi amaçlamıştır. Çalışma; soyutlama, algoritma, örüntü tanıma ve ayrıştırma olmak üzere dört bileşenden oluşan kuramsal bir çerçeveye dayanmaktadır. İlk aşamada 41 maddelik bir havuz hazırlanmış, uzman görüşleriyle içerik geçerliliği sağlanarak 26 maddeye indirgenmiştir. Küçük ölçekli bir pilot çalışmanın ardından, testin son hâli 429 öğrenciye uygulanmıştır. Yapı geçerliliğini incelemek amacıyla Açımlayıcı Faktör Analizi (AFA) ve Doğrulayıcı Faktör Analizi (DFA) gerçekleştirilmiştir. Güvenilirlik analizleri, her alt ölçek için düşük iç tutarlılık (Cronbach's $\alpha = 0{,}30$ - $0{,}50$) ortaya koymuş ve bu durum, her bileşen için daha fazla maddeye ihtiyaç olduğunu göstermiştir. Ayrıca, 2-Parametreli Lojistik Model altında Çok Boyutlu Madde Tepki Teorisi (MIRT) analizi, maddelerin ayırt ediciliği ve güçlüğü hakkında ayrıntılı bilgiler sağlamıştır. Model kabul edilebilir bir uyum göstermekle birlikte, alt ölçekler arasındaki düşük iç tutarlılık, aracın mevcut hâliyle bireysel BİD bileşenlerini ölçmedeki yeterliliğini sınırlamaktadır. Bununla birlikte, çalışma, bütüncül BİD değerlendirmesi için umut verici bir prototip sunmakta ve teorik BİD yapılarının "unplugged" (programlama gerektirmeyen) bağlamlarda psikometrik olarak güvenilir araçlara dönüştürülmesinde karşılaşılan ampirik zorluklar hakkında önemli bilgiler sağlamaktadır. Bulgular, eğitim ortamlarında geçerli ve güvenilir BİD değerlendirme araçlarının geliştirilmesini desteklemek için hem teorik modelin hem de madde havuzunun daha güçlendirilmesi gerektiğini vurgulamaktadır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

α	Cronbach's Alpha: A measure of internal consistency or reliability of a test or scale.

LIST OF ACRONYMS/ABBREVIATIONS

1-PL	One-Parameter Logistics
2-PL	Two-Parameter Logistics
3-PL	Three-Parameter Logistics
EFA	Exploratory Factor Analysis
CFA	Confirmatory Factor Analysis
CFI	Comparative Fit Index
CSTA	Computer Science Teachers Association
CT	Computational Thinking
CTST	Computational Thinking Skills Test
CTT	Classical Test Theory
ICCs	Item Characteristic Curves
I-CVI	Item-Level Content Validity Index
IRT	Item Response Theory
ISTE	International Society for Technology in Education
LM	Lagrange multiplier
MIRT	Multidimensional Item Response Theory
MoNE	Turkish Ministry of National Education
RMSEA	Root Mean Square Error of Approximation
S-CVI/Ave	Scale-Level Content Validity Index based on the average method
TLI	Tucker Lewis Index

1. INTRODUCTION

In today's increasingly digital world, computational thinking (CT) has emerged as a crucial skill set for individuals of all ages, enabling them to approach complex problems logically, systematically, and efficiently. The importance of computational thinking lies in its applicability across various domains, from science and engineering to business and everyday life. Thus, it supports individuals in coping with real-life challenges and in adopting innovations in diverse areas. By developing computational thinking skills, individuals can become more effective problem solvers, critical thinkers, and innovators (Corradini et al., 2017). It is crucial to highlight the significance of computational thinking in today's society and the need to incorporate it into educational curricula to prepare students for the challenges of the digital age (Corradini et al., 2017).

Several studies conducted emphasized the importance of integration of computational thinking into educational curricula (Bers, 2019; Doleck et al., 2017; Peel et al., 2022; Voogt et al., 2015; Wing, 2008). Bers (2019), Relkin et al. (2021), and Strawhacker and Bers (2018) developed new curricula for different grade levels. The Coding as Another Language for KIBO (CAL-KIBO) curriculum, designed by Bers et al. (2022) for preschool and kindergarten students, teaches coding and CT through unplugged activities, including physical interaction with tangible robotics tools such as the KIBO robot. The ABC-Thinking curriculum (Zapata-Cáceres et al., 2024) incorporates Bebras Tasks over 12 weeks. These tasks are categorized into five components: abstraction, algorithmic thinking, decomposition, evaluation, and generalization. The tasks include interactive activities, such as manipulating physical objects, or non-interactive activities that do not require such manipulation. While numerous studies focus on integrating CT through curriculum development, there is a growing need to evaluate how CT is assessed among students—especially in non-programming contexts. Chen et al. (2017) developed an instrument to assess the development of CT skills in fifth-grade students through a new humanoid robotics curriculum they

designed. The curriculum introduces basic computer science concepts and involves students in various projects with robot-based tasks, fostering connections to everyday contexts. In addition to student-level studies, some studies in the teacher-education area focus on preparing educators to integrate CT into their lessons effectively (Ocak et al., 2023; Herro et al., 2022; Otterborn et al., 2019; Umutlu, 2022; Yadav et al., 2016). In addition to curriculum-focused studies, recent research also highlights the importance of teacher preparedness for CT integration. Rich et al. (2019) explored elementary school teachers' understanding of six components of CT—abstraction, algorithmic thinking, automation, debugging, decomposition, and generalization—and their relation with their math and science teaching.

Even though the importance of computational thinking is obvious, there is no consensus on its definition and structure. Numerous theoretical frameworks propose different definitions of CT, most of them focusing on programming-based competencies, while a limited number of them address CT in non-programming contexts. For example, Brennan and Resnick (2012) identify three dimensions for CT: computational concepts, computational practices, and computational perspectives. These dimensions are based on learners' experiences in the programming environment. Similarly, Barr and Stephenson (2011), Grover and Pea (2013), and Repenning et al. (2016) proposed frameworks for CT focusing on programming: context. On the other hand, Wing (2006) emphasized CT use in everyday problem-solving by defining it as a basic skill for everyone, not only for people who work with computers. Csizmadia et al. (2015) advocated for a broad definition of CT, emphasizing that it encompasses cognitive processes used to solve problems beyond the realm of coding. Similarly, the International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) (2011) proposed an expanded definition of CT as a comprehensive problem-solving process that incorporates different skills, not just programming. Despite these efforts, there is still a lack of consensus on defining CT components.

This definitional variability extends to the measurement of CT. Even though there are several assessment tools, most of them prioritize programming skills, partic-

ularly block-based coding. These limit their accessibility and applicability in broader educational contexts. For example, Román-González et al. (2017) attempted to develop a Computational Thinking Test (CTt) and provide evidence for the nature of CT by explaining its associations with key related psychological constructs. Korkmaz et al. (2017) developed a 5-point Likert scale, computational thinking skills (CTS), to assess university students' CT levels based on their self-reported perceptions. Fagerlund et al. (2020) assessed programming contents qualitatively used by fourth-grade students in Scratch projects. On the other hand, Batı (2018) developed a Computational Thinking Test (CTT), which consists of 10 problem case-based questions connected with mathematics and science subjects for 8th-grade students. Although some studies claim to assess CT beyond programming (Wu and Su, 2021), the assessment instruments were not accessible despite multiple attempts to contact the authors. Moreover, these tools lack detailed documentation or evidence of psychometric validation. These gaps highlight the need for a new, accessible, and psychometrically sound instrument.

While several CT assessments exist, there is a need to develop a new scale because existing measurement tools are primarily programming-based, especially block-based coding languages. Also, no accessible scale currently assesses CT skills comprehensively.

In light of these challenges, the present study aims to contribute to a theoretically grounded framework of CT and to develop an assessment tool that measures CT skill of fifth and sixth-grade students without requiring prior programming experience. The result of this study will likely contribute to the existing literature on CT assessment and the improvement of CT education for educators and policymakers.

The following research questions will be used in this study:

(i) What is the underlying factor structure of the Computational Thinking Skills Test (CTST) and does it align with the theoretical framework of computational thinking skills?
(ii) To what extent does the instrument demonstrate construct validity in measuring

the distinct components of computational thinking skills (abstraction, algorithm, pattern recognition, and decomposition) in fifth and sixth-grade students?

(iii) What is the level of internal consistency of the instrument as measured by Cronbach's alpha for each subscale and the overall scale?

(iv) How do items of the Computational Thinking Skills Test (CTST) perform in terms of item difficulty and item discrimination?

1.1. Significance of The Study

With the growing interest and the importance of computational thinking, curricular alignments, the development of courses and materials, and the integration of CT into other disciplines are actions that are taken by nations and countries (Hsu et al., 2018). On the contrary, there are limited studies focused on the assessment of computational thinking skills, especially for middle-grade students (Çoban and Korkmaz, 2021; Li et al., 2021; Román-González, 2015; Shute et al., 2017; Tang et al., 2020). However, it is important to assess students' computational thinking skills in order to understand their level and make appropriate curricular and instructional decisions to improve their computational thinking skills. Therefore, this research holds substantial significance for the field of computational thinking from multiple dimensions.

According to Shute et al. (2017), the lack of a widely accepted CT assessment causes a major problem in the field, which is the comparison of CT results from different countries. This study may contribute to the existing literature on CT by being a reliable and valid assessment instrument for middle-grade students' CT skills. Additionally, the current study may provide an important contribution to the field by developing and validating an assessment that measures specific dimensions (abstraction, algorithm, pattern recognition, and decomposition) of CT skills in fifth and sixth-grade students.

Tang et al. (2020) asserted that research on CT assessment for workshops on professional development and teacher education was scarce. Moreover, Kang et al. (2018) found that teachers scored the lowest in SEP (science and engineering practice)

5, which involves using mathematics and computational thinking in terms of both knowledge and confidence. Therefore, developing an assessment to measure CT skills in middle-grade students may indirectly inform the design of professional development programs for teachers. Although this instrument does not directly assess teachers' CT knowledge, the results may highlight areas where students struggle—thereby informing the design of teacher professional development in CT integration.

Moreover, CT can be applied in a wide range of learning environments beyond just computer science (Kalelioğlu et al., 2016). In her definition, Wing (2008) also emphasized, "Computational thinking is a fundamental skill for everyone, not just for computer scientists" (p.33, 2006). CT can be used to strengthen students' comprehension in different fields, such as STEM, non-STEM, and daily-life problems (ISTE and CSTA, 2011; Weintrop et al., 2016; Wing, 2008). Furthermore, Grover and Pea (2018) argue that the transfer of thinking skills like CT to other domains, such as STEM, cannot be achieved directly. In order to facilitate this transfer, there is a need for proven methods that clearly connect the original and the new learning contexts. However, despite the growing need to integrate computational thinking (CT) into non-STEM areas, Tang et al. (2020) argued that assessments combining CT with non-STEM subjects remain underdeveloped. As current CT assessment concepts are primarily focused on computing and programming contexts, this may discourage the use of CT skills for those unfamiliar with such environments (Tang et al, 2020). Therefore, this instrument may contribute to the assessment of CT skills of fifth- and sixth-grade students from diverse backgrounds, enabling them to effectively apply these skills across various domains without requiring a computing or programming background.

The results of this study may shed light on the development of a validated CT assessment for fifth and sixth-grade students by measuring specific dimensions: abstraction, algorithm, pattern recognition, and decomposition. This may assist educators and researchers in understanding students' strengths and weaknesses in CT skills and designing targeted instructional practices, interventions, and curricula to improve their CT skills. Also, this instrument will fill the gap in the existing literature, which

is an assessment of CT skills without the need for familiarity with computing or programming. Thus, the findings will contribute to the development and assessment of CT in middle schools in Turkey, as it forms the basis of the Information Technologies and Software course of the Turkish Ministry of National Education (MoNE, 2023).

1.2. Purpose of The Study

This study aims to develop an instrument to assess fifth and sixth-grade students' computational thinking skills. The specific dimensions used in this assessment will be abstraction, algorithm, pattern recognition, and decomposition.

2. LITERATURE REVIEW

2.1. Computational Thinking

In his book "Mindstorms: Children, Computers, and Powerful Ideas," Papert (1980) discussed the effect of computers on people's thinking and learning. He asserted that the use of computers contributes to cognitive processes, not just by serving practical functions but by reshaping conceptual thinking, even in the absence of direct physical interaction with technology, such as computers (Papert, 1980). Moreover, he designed the LOGO programming language to observe the transformative potential of programming to enhance thinking.

Even though Papert (1980) addressed computational thinking in his book, Jeannette Wing (2006) was the first to use the term "computational thinking" as "a fundamental skill for everyone, not just for computer scientists" (p.33). Therefore, it can be used everywhere (Wing, 2008), so it should be learned by every child, like reading, writing, and arithmetic (Wing, 2006). In 2011, she enhanced her definition of CT as "the thinking processes involved in generating a problem and finding its solution(s) in such a way that a computer-human or machine can effectively carry out." Similarly, Aho (2012) defined CT as the mental processes of framing problems so their solutions can be expressed as computational steps and algorithms. The International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) expanded the definition of it as a problem-solving process, which includes such traits: developing challenges that can be solved with the aid of a computer and other resources, analyzing and arranging data systematically, representing data by using abstractions, creating automated solutions by using algorithmic thinking, which involves designing a structured sequence of steps, identifying, assessing, and implementing potential solutions to achieve the most efficient and effective use of actions and resources, and the ability to generalize and apply this approach to a wide range of problems (2011). CT is a thought process for problem-solving in systematic ways that can be

computerized and can be transferred across various subjects, such as "quantum physics, advanced biology, human-computer systems, and development of useful computational tools" (Cuny et al., 2010; p.51, Barr and Stephenson, 2011). These definitions reveal that a CT is an everyday skill that every child should learn throughout their education.

In parallel with this idea, several frameworks and international or national curricula have integrated computational thinking in recent years (Angeli et al., 2016; Barr and Stephenson, 2011; Bers, 2019; Brennan and Resnick, 2012; Kalelioğlu et al., 2016; Uzunboylu et al., 2017). While various frameworks include additional skills such as debugging or generalization, this study focuses on abstraction, algorithm, pattern recognition, and decomposition as they are consistently emphasized across foundational models (e.g., Wing, 2006; ISTE and CSTA, 2011; Csizmadia et al., 2015) and are developmentally appropriate for middle-grade learners. For instance, Angeli et al. (2016) developed a CT curriculum framework for K-6 based on the five skills: abstraction, generalization, decomposition, algorithmic thinking, and debugging through computer programming. Furthermore, Barr and Stephenson (2011) formed a more comprehensive framework comprising nine CT concepts or capabilities: data collection, data analysis, data representation, problem decomposition, abstraction, algorithms and procedures, automation, parallelization, and simulation, linking them with subject areas through examples.

Besides, Bers (2012) developed a Positive Technological Development (PTD) framework focusing on personal strengths, technology-based actions, and practice. Personal strengths - based technology is defined as the six C's: competence, confidence, character, caring, connection, and contribution (Bers, 2012). Similarly, Brennan and Resnick (2012) developed a CT framework based on Scratch, a programming tool, encompassing three dimensions: computational concepts, computational practices, and computational perspectives. These dimensions respectively address the concepts utilized during programming, the methods employed, and the perspectives formed by the designer regarding both the external world and their personal experiences. Grover and Pea (2013) outlined the key elements of CT that underpin its learning and assessment of

its development, serving as the foundation for curricula. The elements include abstraction and pattern generalization, systematic information processing, representations, algorithmic flow control, structured problem decomposition, iterative, recursive, and parallel thinking, conditional logic, debugging, and efficiency considerations (Grover and Pea, 2013). Kalelioğlu et al. (2016) proposed a framework that combines problem-solving and CT skills, organizing them into five key categories. These include identifying the problem; gathering, representing, and analyzing data; generating, selecting, and planning solutions; implementing the solutions; and assessing and continuing them for improvement (Kalelioğlu et al., 2016). Under each category, many actions, such as abstraction, data collection, automation, mathematical reasoning, parallelization, and debugging, are selected from an analysis of papers, the operational definition of CT, and their findings (Kalelioğlu et al., 2016). Despite the various approaches and frameworks to CT in the literature, there is no consensus on the set of skills or elements of CT. Thus, this study adopts the four-component model—abstraction, algorithm, pattern recognition, and decomposition—based on their recurrence across key CT frameworks (e.g., Wing, 2006; ISTE and CSTA, 2011; Csizmadia et al., 2015). These components were chosen due to their cognitive accessibility for middle-grade students and their relevance to cross-disciplinary problem-solving.

In addition to frameworks, the integration of CT into curricula has been explored in various studies. Chen et al. (2017) investigated the assessment of CT skills of elementary students through their reasoning skills of everyday scenarios and robotics programming curriculum. On the other hand, using PTD, the TangibleK Robotics Program, and the Coding as Another Language (CAL) curriculum with the KIBO robotics kit or ScratchJr are developed for the acquisition of CT skills in preschool and kindergarten students with tangible robotics tools (Bers et al., 2014; Bers, 2019). Bers (2019) identified six developmental coding stages that children progress through when learning with the CAL curriculum. These stages are emergent, coding and decoding, fluency, new knowledge, multiple perspectives, and purposefulness (Bers, 2019). Furthermore, the Foundations for Advancing Computational Thinking (FACT) curriculum aimed to enhance understanding and transfer for middle grades through structured and

open-ended Scratch programming assignments (Grover, 2017). Zapata-Cáceres et al. (2024) formed an ABC-Thinking curriculum, which includes various Bebras tasks classified under CT skills: abstraction, algorithmic thinking, decomposition, evaluation, and generalization for 12 weeks. The ABC-Thinking curriculum differs from other studies in the literature by encompassing Bebras Tasks, which is an assessment instrument and also a learning tool without needing any programming environment or computer-related tool (Zapata-Cáceres et al., 2024). By considering CT curriculum examples from the literature, the curriculum examples in existing literature focus on teaching CT through programming and computer-based tools. However, there is a scarcity of studies that aim to measure CT skills without using programming or digital tools.

Besides frameworks and curricula developments in the CT area, various assessment approaches have been proposed in the literature regarding the measurement of CT skills. It is essential to develop a valid assessment tool for evaluating the effectiveness of CT curricula (Grover and Pea, 2013). A valid and suitable assessment tool can enable teachers to evaluate their lessons and curricula (Relkin et al., 2023). Similar to the lack of consensus on the definition of CT, there is no universally accepted measurement tool for CT skills. A wide range of CT assessments exists, which vary in terms of skill components, modality (programming-based vs. unplugged), and target age group. In the following sections, key tools are reviewed under two main categories: assessments requiring programming and those that do not.

Building on previously reviewed assessment approaches, this section focuses specifically on tools that emphasize programming environments and their limitations. Román-González (2015) developed the Computational Thinking Test, a multiple-choice test for assessing the CT skills of K-7 and K-8 Spanish students. The test items address eight computational concepts: basic directions, loops repeat times, Loops – repeat until, if – simple conditional, if/else complex conditional, while conditional, simple functions, and functions with parameters, which are derived from programming languages (Román-González, 2015). Moreover, Chen et al. (2017) developed an instrument that

employed both multiple-choice and open-ended questions to assess the CT skills of fifth graders in two contexts: everyday scenarios and robotics programming. Based on the CSTA framework (2011), excluding the transfer of skill aspect, the study focused on these components: syntax, data, algorithms, representation, and achieving efficiency and effectiveness (Chen et al., 2017). In addition to the Chen et al. (2017) instrument, other assessments have emerged that evaluate CT through programming environments. For instance, the Fairy Assessment is a performance assessment formed in the Alice game programming environment and aims to measure the CT skills of middle school students (Werner et al., 2012). Similarly, Dr. Scratch is a web-based programming application that automatically assesses the appropriateness of Scratch programs developed by students (Moreno-León et al., 2015). The application evaluates the competence level of students through seven CT concepts: abstraction and problem decomposition, parallelism, logical thinking, synchronization, flow control, user interactivity, and data representation (Moreno-León et al., 2015). Bebras International Contest on Informatics and Computational Thinking tasks are used to assess CT skills in three age groups (ages 11-14, 15-16, and upper secondary level) through multiple choice questions and computer-related interactive tasks (Dagienė and Futschek, 2008). On the other hand, there are CT scales that assess the perceptions or attitudes toward computational thinking skill, such as the Computational Thinking Scales (CTS) (Korkmaz et al., 2017) and the Computational Thinking Skills Scale (CTSS) (Durak and Saritepeci, 2018).

In addition to these computer or programming-related assessments and attitudes-perceptions scales, only a few studies have attempted to develop assessment tools that do not require computer or programming skills. Mindetbay et al. (2019) developed a Computational Thinking Performance Test, including 50 multiple-choice questions from logical thinking, abstraction, and generalization concepts to assess eighth-grade students' CT performances. TechCheck Assessment is an unplugged assessment tool for early childhood without requiring any prior programming skills (Relkin et al., 2020). The assessment covers appropriate domains for young ages defined by Bers (2020): algorithms, modularity, control structures, representation, hardware/software, and de-

bugging, except for the design process (Relkin et al., 2021). To provide a structured overview of existing CT assessments, the tools are categorized based on the CT components they measure, the target age group, the type and number of items included, and their accessibility. The reasons why each tool may not be suitable for this study, such as misalignment with the four-component model, which will be explained later, an age-level mismatch, or limited access to the tool, are detailed in the notes column (see Table 1 in Appendix A).

Overall, the existing literature presents various assessment tools for computational thinking. However, most of these assessments primarily emphasize programming or computer-related concepts. Although TechCheck (Relkin et al., 2020) is described as an unplugged CT assessment for young children, the assessment includes computer-related concepts, such as modularity, control structures, hardware/software, and debugging. Similarly, several tools such as Fairy Assessment (Werner et al., 2012), Computational Thinking Test (CT-test 2.0) (Román-González, 2017), Computational Thinking Challenge (CTC) (Lai, 2021), and Chen et al. (2017)'s instrument are based on programming. Shen et al. (2024) recently developed the Computational Thinking Assessment for Elementary Students (CTAES) for grades 3–5. While the instrument demonstrated promising psychometric properties through Rasch analyses, its item content still integrates programming-related elements, such as loops, conditional statements, variables, and debugging, limiting its applicability in non-programming contexts. On the other hand, in several assessments, such as the Computational Thinking Assessment (CTA-CES) (Li et al., 2021) and The Computational Assessment (Wu and Su, 2021), access to the full instrument or sample items was not granted despite multiple contact attempts. Including programming elements in content or lacking transparency limits the assessment's applicability in non-programming contexts. Consequently, there is a need for a CT assessment tool that measures students' CT skills without relying on programming or computer-related concepts, particularly for fifth and sixth-graders. Additionally, the majority of these assessments are available only in English, highlighting the necessity for Turkish-language assessment tools to support curriculum development.

To ensure that the assessment effectively measures computational thinking (CT) skills without relying on programming or computer-related concepts, it is crucial to identify the key CT components that will guide its development.

2.2. A Four-Component Model of CT

This study adopts a four-component model of computational thinking (CT), comprising abstraction, algorithm, pattern recognition, and decomposition. This model is grounded in the recurrence of these components across widely recognized CT frameworks (e.g., Wing, 2006; ISTE and CSTA, 2011; Csizmadia, 2015). It reflects core cognitive skills that are essential to interdisciplinary problem-solving.

To synthesize the model, existing CT frameworks proposed in the literature were systematically analyzed to identify their core components. Table 2 was formed according to the findings from these studies (see Table 2 in Appendix B). Due to the diversity of the components used across the literature, components with similar or overlapping meanings were identified and grouped under the same component. For instance, Selby and Woollard (2014) defined generalization as the ability to identify and reuse common patterns when solving a problem, which closely aligns with the component of pattern recognition. Consequently, generalization, included as a CT component by Angeli et al. (2016), Kalelioğlu et al. (2016), Csizmadia (2015), Wing (2006), and Selby and Woollard (2014), is represented as pattern recognition in Table 2.

Among these identified components in the articles, the most frequently cited across frameworks were chosen. These include abstraction, decomposition, pattern recognition, algorithm, automation, evaluation, parallelization, testing and debugging, data collection, data analysis, data representation, and simulation. However, not all of these were incorporated into the proposed four-component model.

According to Selby and Woollard (2014), automation is recognized in several studies (e.g., Barr and Stephenson, 2011; Grover and Pea, 2013; ISTE and CSTA, 2011;

Repenning et al., 2016; Kalelioğlu et al., 2016; Wing, 2006) as the process by which requires a computer to run a program; it was excluded from our four-component model in this study because CT is a thought process, not necessarily reliant on programming or computer-based tools. Moreover, evaluation is defined as the confirmation process of the solution, whether it meets the intended purpose or not (Csizmadia, 2015). However, the evaluation is a component that should not be separated from other components of CT, including abstraction, algorithm, pattern recognition, and decomposition. During the process of each component, students should evaluate whether the solution to the problem is appropriate or not for meeting the purpose. So, rather than taking evaluation as a separate component of CT, it should be integrated into the process of CT.

On the other hand, parallelization, which is defined as the dividing of tasks or data to solve them simultaneously, is taken as a component of CT in the literature (Barr and Stephenson, 2011; ISTE and CSTA, 2011). Similarly, testing and debugging are used as another component to define the CT framework, meaning running a solution to detect and correct errors (Angeli et al., 2016). However, these two components, parallelization and testing and debugging are related to computer science and generally taken in the programming context rather than to generalizable CT processes. For this reason, they were excluded from the present model.

Likewise, data-related components—data collection, data analysis, data representation, and simulation (ISTE and CSTA, 2011) are rarely emphasized across CT frameworks. Data collection means collecting data, data analysis is defined as "making sense of data, finding patterns, and drawing conclusions," and data representation means representing the data through visuals, graphs, or tables (ISTE and CSTA, 2011, p.14). Similarly, simulation means representing the procedure (ISTE and CSTA, 2011). These components often support CT development but are not considered core to the cognitive essence of CT itself (Selby and Woollard, 2014). Hence, they are omitted from the four-component model.

Other components, such as incremental and iterative thinking, expressing, connecting, and questioning, reusing and remixing, collaboration and creativity, logical thinking, conceptualizing, and mathematical reasoning appear less frequently and focus on general concepts rather than distinct CT components. As such, they are also excluded.

In conclusion, this study proposes a four-component model of CT consisting of abstraction, algorithm, pattern recognition, and decomposition. These components are not only consistently supported in the literature but are also developmentally appropriate for middle-grade students and are relevant to cross-disciplinary problem-solving. The model establishes a balance between theoretical depth and practical application. It is developmentally appropriate, adaptable to both digital and unplugged learning environments, and applicable across various subject areas. Furthermore, its clarity and coherence also support effective teaching, assessment, and curriculum design. By focusing on these four interrelated yet distinct components, the model offers a strong and accessible framework for developing CT skills in K–12 education.

2.2.1. The Components of CT

The International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) (2011) developed an operational definition for CT by listing concepts such as data collection, data analysis and representation, problem decomposition, abstraction, algorithms and procedures, automation, simulation, and parallelization. According to their literature review study, Kalelioğlu et al. (2016) found the most common dimensions of CT as abstraction, problem-solving, algorithmic thinking, pattern recognition, and design-based thinking. Cansu and Cansu (2019) outlined key components of computational thinking (CT) as identified by various researchers, including abstraction, automation, and analysis (Lee et al., 2011); abstraction, algorithmic thinking, decomposition, evaluation, and generalization (Selby and Woollard, 2014); abstraction, algorithms, decomposition, debugging, and generalization (Angeli et al., 2016); and abstraction, algorithms, automation, problem decompo-

sition, and generalization (Wing, 2006, 2008, 2011). According to these components, Cansu and Cansu (2019) chose five: abstraction, problem decomposition, algorithmic thinking, automation, and generalization, which are also defined by Csizmadia et al. (2015). Selby and Woollard (2014) emphasize the importance of including the concept of a thought process in defining computational thinking. In summary, despite diverse categorizations of CT skills in the literature, abstraction, algorithm, pattern recognition, and decomposition are foundational components present in most frameworks and adaptable to unplugged, non-programming tasks. Therefore, they were selected to structure the assessment in this study.

2.3. Definitions of The Components of The CT

2.3.1. Abstraction

Wing (2006) defines computational thinking as using abstraction and decomposition to solve or design large, complex tasks. She emphasizes that abstraction and decomposition are fundamental processes in computational thinking, among several others, including pattern recognition and algorithmic thinking. In 2011, Wing expanded her definition of abstraction, emphasizing that the most crucial and advanced process involves identifying patterns, generalizing from specific examples, and creating parameters. Through abstraction, a single object can represent multiple instances by concentrating on the essential characteristics shared by a group while disregarding irrelevant differences. ISTE and CSTA define abstraction as ”reducing complexity to define the main idea” (2011, p. 14). While solving problems, the process of simplifying from specific to general is an abstraction. (Barr and Stephenson, 2011). Likewise, Csizmadia et al. (2015) define abstraction as simplifying a problem or system by removing unnecessary details to make it more understandable. For instance, while shopping on the Internet, using filters to see only the relevant attributes like price, brand, or color is an example of abstraction. In light of these definitions, abstraction refers to the process of ignoring unnecessary details and concentrating on the critical aspects of a problem or system to make it more understandable and manageable.

2.3.2. Algorithm

Wing (2008) defines abstraction as an algorithm that takes an input and turns it into a desired output in a way that is implemented step by step. Aho (2012) emphasizes representing the solutions to problems as computational steps and algorithms in the CT process. Thus, computational thinking should include algorithms as one of its components. An algorithm is a process that takes inputs, follows a set of steps, and produces outputs to reach a specific goal (Wing, 2011). In addition, ISTE and CSTA (2011) define algorithms and procedures as a "series of ordered steps taken to solve a problem or achieve some end" (2011, p.15). The term algorithm is understood as a systematic, step-by-step process for completing tasks, not only in computer science but also in various other fields (Selby and Woollard, 2014). For instance, following the steps of a recipe for baking a cake demonstrates algorithm usage. The algorithm is the ability to solve problems through a series of ordered steps.

2.3.3. Pattern Recognition

Wing (2011) uses algorithmic and parallel thinking in her definition of computational thinking: algorithmic and parallel thinking involve other thinking processes, such as compositional reasoning, pattern matching, procedural thinking, and recursive thinking. Also, the most crucial and high-level component of CT, abstraction, includes the identification of patterns (Wing, 2011). Grover and Pea (2013) consider pattern generalization to be an integral element of CT. Shute et al. (2017) summarize pattern recognition as recognition of the underlying patterns and principles that form the foundation of the data and information's structure under the abstraction component of CT. Computational thinking skills can be developed with or without computers by analyzing processes, identifying real-life problems, identifying data patterns, and evaluating evidence critically (Charlton and Luckin, 2012). Csizmadia et al. (2015) include generalizations in CT concepts by determining patterns, similarities, and connections and implementing prior experiences to new ones. Abdullah et al. (2019) defined pattern recognition as "identifying the structure and pattern of a problem and finding

similarities between current information and past information" (p. 2). They asserted that it is necessary to recognize similarities between past and current problems to find solutions in pattern recognition (2019). For example, somebody can adjust the ratio of flour to water in the bread recipe according to the amount of bread she wants to bake. According to the literature, pattern recognition is one of the key components of computational thinking, which can be defined as identifying and extracting similarities from given information to solve problems easily.

2.3.4. Decomposition

Wing (2006) defines computational thinking as using abstraction and decomposition to solve large and complex problems. The NRC Report (2010) includes decomposition in the CT definition. ISTE and CSTA define decomposition as "breaking down tasks into smaller, manageable parts" (2011, p. 14), similar to Barr and Stephenson's (2011) definition of "breaking problems down into smaller parts that may be more easily solved" (p. 52). Selby and Woollard (2014) also highlight decomposition as a key component of CT, defining it as the integration of thought processes that focus on problem-solving.

Decomposition involves thinking of problems or complex situations by dividing them into their components to deal with them easily (Csizmadia et al., 2015). The decomposition breaks down a large problem into smaller, more manageable parts, making it easier to solve (Abdullah et al., 2019). Similarly, Shute et al. (2017) explain that decomposition involves breaking down complex systems or objects into smaller, more manageable parts. In parallel with the existing literature, decomposition can be defined as the ability to break down a complex problem or task into smaller, more manageable parts to solve them. For instance, a child can use decomposition when cleaning their room by breaking the task into smaller, manageable steps like picking up toys, organizing books, putting away clothes, and making the bed.

3. METHOD

3.1. Sampling and Participants

Participants were selected through the convenience sampling method because of the availability and accessibility within the participating school districts. Fifth and sixth graders (10-11 ages) were specifically chosen for this study based on the cognitive levels targeted by the Bebras Tasks (Dagienė and Futschek, 2008). Bebras tasks are categorized into three age groups: Benjamin (grades 5–8), Junior (grades 9–10), and Senior (grades 11–13). Some tasks may be appropriate for more than one age group, although difficulty levels can differ. The Benjamin group, which includes fifth and sixth-graders, aligns with the cognitive demands of most tasks in the study, making this age group suitable for assessing computational thinking skills in a non-programming context. The selected age group corresponds to a developmental stage in which students' cognitive abilities are sufficiently mature to engage with these types of problems, making them suitable participants for the assessment of computational thinking in a non-programming context. Given that the item pool includes 26 items, a sample size of 430 students is targeted to meet the 5–10 participants-per-item guideline (Nunnally, 1967).

To ensure the clarity and appropriateness of the items before large-scale implementation, a pilot study was conducted in March 2025 with 12 students, five from the 5th grade and seven from the 6th grade. The test was administered by a teacher working in a rural public school in İzmir due to accessibility. The primary aim of the pilot study was to evaluate whether the items were comprehensible to students in terms of wording, readability, and clarity of instructions, as well as to check the suitability of the difficulty level and the time required to complete the test. During the administration, the teacher reported that most students struggled to solve the questions and answered randomly. Moreover, the allocated time was not sufficient for many of them, but the teacher emphasized that even if more time had been given, most students would not

have been able to complete the test due to their relatively low academic performance levels. Notably, even the successful student in the class was only able to answer a portion of the items.

Based on the teacher's observations, it was considered that the pilot students' difficulties primarily reflected their lower academic background rather than an inherent problem with the test design. Therefore, while the items were found to be challenging for this group, expert opinions and the teacher's feedback supported that the difficulty level was appropriate for the intended target population. Consequently, no adjustment was made to the overall difficulty of the test, though one item was slightly revised for improved readability before proceeding to the final administration.

During the final test phase, efforts were made to reach a more diverse and larger sample. Data were collected from two public and one private school across two districts in Istanbul, as well as from one public school in Kocaeli. The required permissions for data collection were taken from the Ministry of National Education. The final test was administered to 201 fifth-grade students and 227 sixth-grade students; two students did not specify their grade level. One participant was excluded from the analysis after self-reporting that all questions were answered randomly.

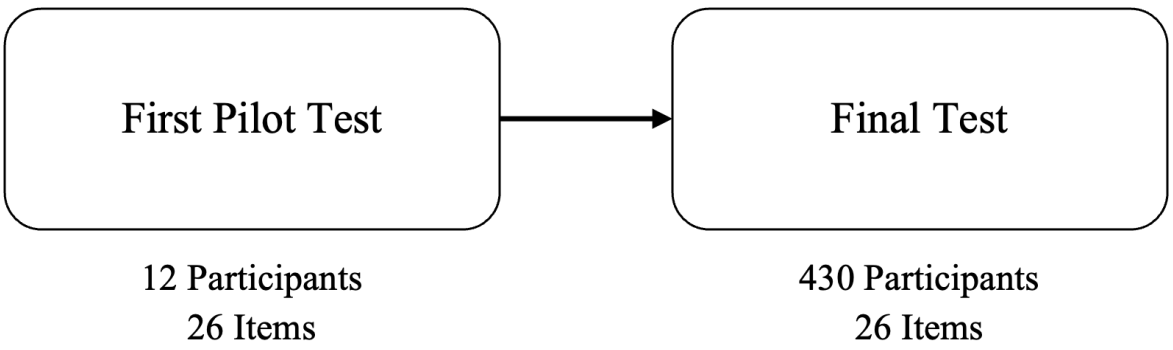


Figure 3.1: Test Implementation Overview.

3.2. Instrument

The main aim of this study is to develop a test to assess fifth and sixth-grade students' computational thinking skills. There were several steps to develop the instrument. The process of instrument development is given in Figure 3.1.

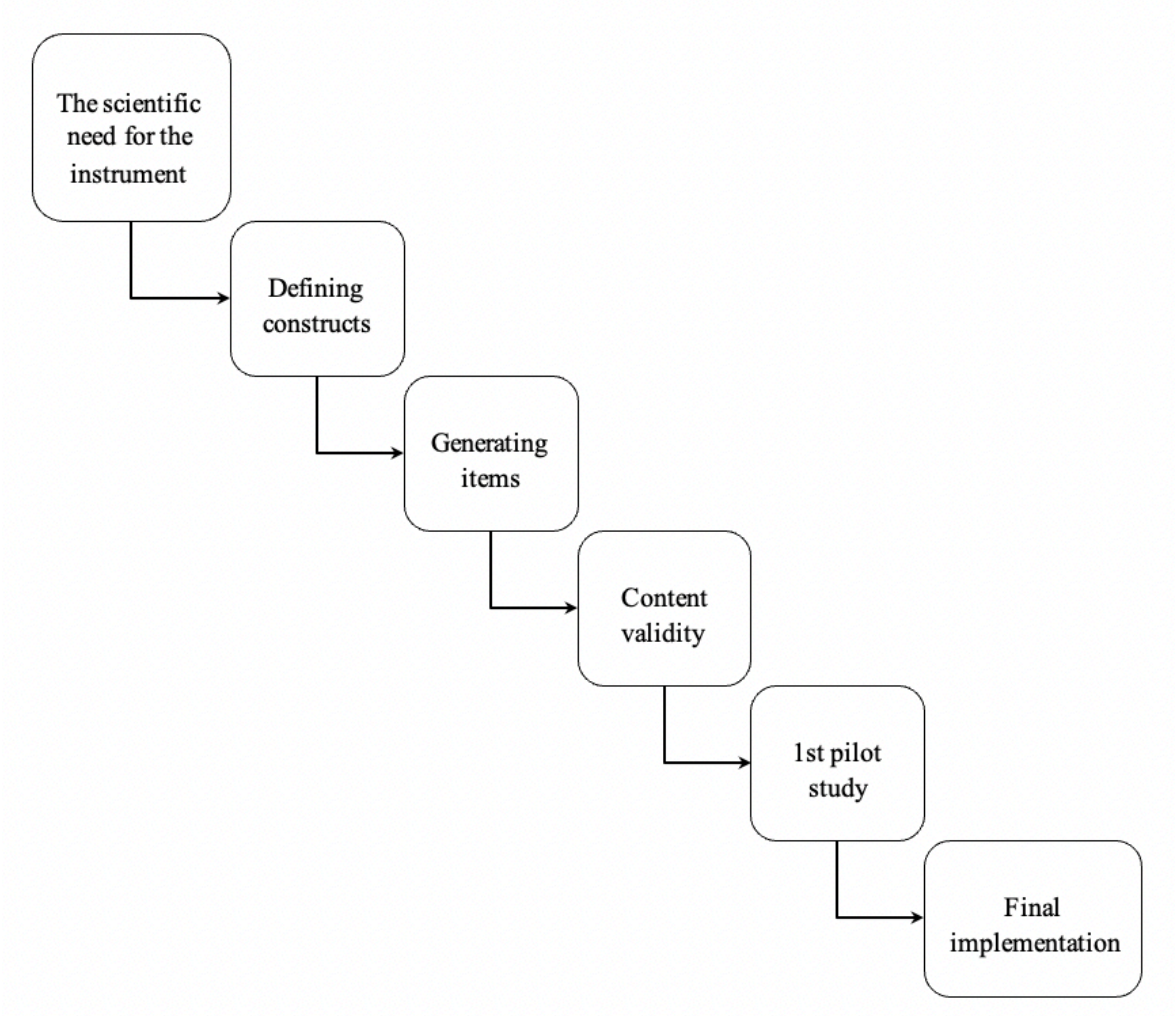


Figure 3.2: The Instrument Development Process.

3.2.1. The Scientific Need for The Instrument

Holmbeck and Devine (2009) stated that identifying a clear scientific need is an essential part of developing a new tool. Developers should determine how the

tool differs from other tools and the reason it is necessary for a specific population (Holmbeck and Devine, 2009).

Although various CT assessment tools exist, a significant need for a new scale remains because existing instruments are largely centered around programming, particularly block-based coding, which limits their applicability. Additionally, there is no widely available tool that evaluates computational thinking (CT) comprehensively across its four core components: abstraction, algorithm, pattern recognition, and decomposition, without assuming prior coding experience. While some studies, such as Wu and Su (2021), claim to measure CT beyond coding, the associated tools were unavailable despite repeated efforts to obtain them. Furthermore, these instruments often lack clear documentation or evidence of psychometric reliability and validity. These limitations emphasize the need for a new, accessible, and rigorously validated CT assessment tailored to fifth- and sixth-grade students. Therefore, a Computational Thinking Skills Test (CTST) was developed as part of this study to measure the computational thinking skills of fifth and sixth-grade students.

3.2.2. Defining Constructs

It is important to define the constructs clearly and their underlying dimensions to measure the intended assessment (Arpacıoğlu, 2023; Holmbeck and Devine, 2009). A clearly defined construct establishes the limits of the construct and simplifies the item development and content validation process (Tosun, 2019). A comprehensive review of existing frameworks and assessments for computational thinking guided the item development. Even though different frameworks incorporate additional components, there is no agreement on a standardized set of CT skills. Therefore, this study adopts the four components —abstraction, algorithm, pattern recognition, and decomposition as constructs—based on their recurrence across key CT frameworks (e.g., Wing, 2006; ISTE and CSTA, 2011; Csizmadia et al., 2015).

3.2.3. Generating Items

The item generation process is important to ensure that the items adequately represent the constructs being measured. Items were generated to represent the four computational thinking components: abstraction, algorithm, pattern recognition, and decomposition. The initial pool was formed of 41 multiple-choice and scenario-based items to assess students' CT abilities. Approximately 10-12 items were developed for each component. Thus, an item pool 1.5 to 2 times the size of the final test was developed following psychometric recommendations to allow for revisions and item elimination after pilot testing (Nunnally and Bernstein, 1994). While no items were directly adapted, the Bebras Challenge tasks, widely used to evaluate the computational thinking skills of students from various grade levels, served as a model for the scenario-based item style used in our instrument (Dagiene and Futschek, 2008). However, unlike the Bebras Challenge Tasks' items, where multiple CT components were assessed in one item, each item in our instrument was deliberately designed to assess only one CT component in order to allow for clearer construct representation and factor analysis. Therefore, all items were written by the author. After item eliminations and revisions, in terms of clarity and readability, 26 items remained in the final version.

3.2.4. Content Validity

The initial version of the test was submitted to eight experts specializing in educational technology, computer science, and assessment and evaluation. These experts were asked to evaluate the items in terms of content validity, clarity of wording, appropriateness of the item to the test and age of participants, difficulty level (categorized as easy, medium, or hard), and alignment with computational thinking constructs (abstraction, algorithm, pattern recognition, and decomposition) to ensure that the test measures the intended abilities accurately (Haladyna et al., 2002).

Additionally, the experts were invited to provide written feedback on each item. To assess content validity, the Item-Level Content Validity Index (I-CVI) was calcu-

lated, representing the proportion of experts who rated each item as appropriate (Lynn, 1986). Similarly, the Scale-Level Content Validity Index based on the average method (S-CVI/Ave) was calculated for the expert ratings for the overall scale (Polit et al., 2007).

Items with an I-CVI score of 0.80 or higher were considered acceptable, resulting in the selection of 24 questions (Lynn, 1986). Two additional items sharing the same scenario as the selected two were also drawn from the item pool. The selected items were revised based on the experts' comments to improve clarity and content validity. Specifically, items 3, 5, 6, 7, 8, 11, 12, 14, 16, and 21 were revised to enhance the clarity of their content, visuals, or tables. Item 1 was revised for cultural adaptation and according to the target age level, because it required knowledge of brewing tea in Turkish culture. Items 2, 9, 13, 18, 19, and 22 were revised by removing unnecessary information to concise and focused wording. The visuals in items 20 and 25 were revised to make them more understandable. In item 15, a table was used instead of descriptive text of books to reduce cognitive load and maintain student attention.

(a) Original version of the question

(b) Revised version of the question

Figure 3.3: The original version (a) and the revised version (b) of the question for clarity following the pilot study.

The difficulty level of each item was determined by calculating the mean of the ratings provided by the experts, where difficulty levels were scored as follows: 1 for

easy, 2 for medium, and 3 for hard. The mean difficulty score was calculated for each item and for the overall test, yielding a total mean difficulty score of 1.69. Based on this value, difficulty ranges were established as follows: 1.00–1.65 (easy), 1.66–2.32 (medium), and 2.33–3.00 (hard). Accordingly, the items in the final version of the test were organized in ascending order of difficulty, from easy to hard.

Table 3.1: The difficulty of the items.

	Easy (1.00–1.65)	**Medium (1.66–2.32)**	**Hard (2.33–3.00)**
The item number ranges	1 to 12	13 to 25	25 to 26

3.2.5. Pilot Study

The final version of the Computational Thinking Skills Test (CTST) consisted of 26 items, distributed across four core constructs: 9 items for abstraction, 5 for algorithm, 6 for pattern recognition, and 6 for decomposition (see Table 3.2). Each item in the CTST was categorized under a single CT component to ensure clarity in measurement and the validity of the factor structure.

Table 3.2: The distribution of the items among CT components in the final version.

	Abstraction	**Algorithm**	**Pattern Recognition**	**Decomposition**
Item numbers	3, 5, 7, 8, 10, 13, 19, 21, 26	1, 6, 9, 14, 24	2, 11, 16, 17, 20, 25	4, 12, 15, 18, 22, 23
Total items	9	5	6	6

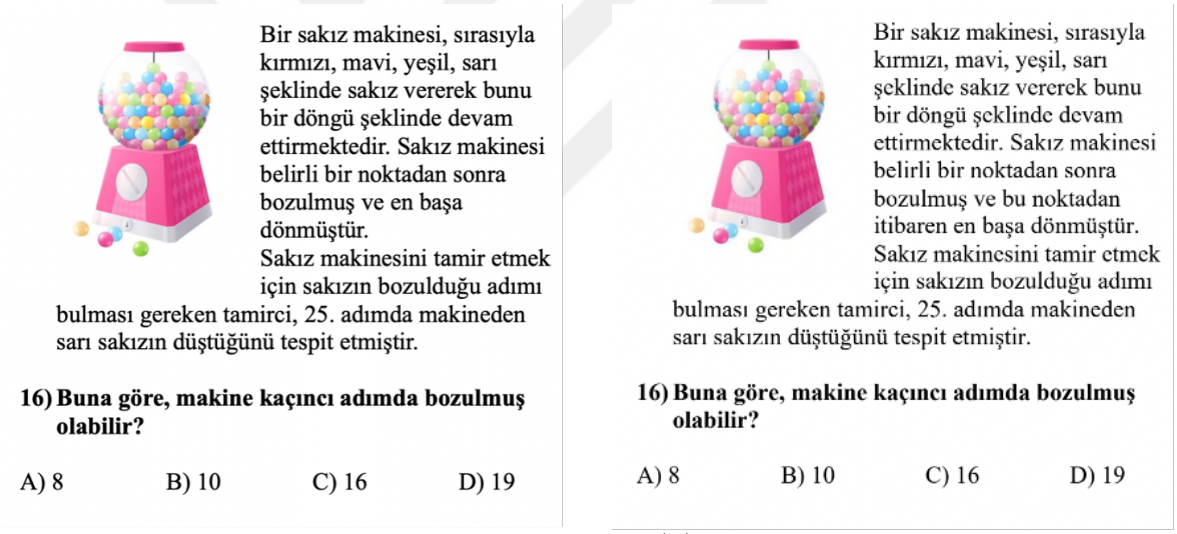
The CTST was administered to a small sample of fifth- and sixth-grade students, comprising five students from the fifth grade and seven from the sixth grade. These

students did not participate in the main study.

This pilot study aimed to assess the following aspects:

- Item clarity and readability
- Item difficulty levels
- Time required for test completion

Based on the overall students' performance and verbal feedback from their teachers, a revision was made to one item to improve its clarity (see Figure 3.4).



(a) Original version of the question (b) Revised version of the question

Figure 3.4: The original version (a) and the revised version (b) of the question for clarity following the pilot study.

It was estimated by the researcher that the test could be completed in approximately 40 minutes. However, the students could not complete the test within the expected time based on the teacher's feedback. So, more time was required to complete the test. Given the limited sample size of the pilot study, a formal quantitative analysis of item difficulty was not conducted at this stage. However, preliminary qualitative feedback from the administering teacher indicated that students experienced

significant challenges with the items, suggesting a higher-than-anticipated difficulty level for the target population. The administering teacher reported that students were unable to complete most of the items, noting that the questions were too advanced for their level and that the pilot group consisted of generally lower-performing students. Based on this observation, the researcher concluded that the students' inability to complete the items was likely due to the mismatch between item difficulty and the students' current skill level. After the pilot study, a minor revision was made by rephrasing one question to improve its clarity.

The next step was to conduct a final implementation with a larger sample to proceed with item and reliability analyses.

3.2.6. Final Implementation

The final implementation phase prioritized securing a larger sample. To this end, data were collected from a convenience sample of schools: two public and one private school across two districts in Istanbul, and one public school in Kocaeli. The necessary permissions to collect data from students were obtained from the Ministry of National Education.

The revised CTST, which was revised from the pilot version by rephrasing one question for improved clarity, was administered to 201 fifth-grade and 227 sixth-grade students. The aim of this final implementation is to examine the psychometric properties of the instrument more comprehensively. Item analysis was conducted to examine item difficulty, item discrimination, and item-total correlations. Internal consistency was assessed using Cronbach's alpha for both the overall scale and each subscale.

Construct validity was evaluated through both Exploratory Factor Analysis (EFA) and Confirmatory Factor Analysis (CFA) to verify alignment with the four components of computational thinking. The results of this main study provided the final evidence needed to establish the validity and reliability of the CTST.

3.3. Data Analysis

The data analysis in this study followed several steps to check the validity and reliability of the Computational Thinking Skills Test (CTST) designed for 5th and 6th-grade students. Content and construct validity, reliability, and item analysis were assessed to confirm the validity and reliability of the test.

Content validity was examined through the Item-Level Content Validity Index (I-CVI) by expert opinion forms, while construct validity was evaluated using Exploratory Factor Analysis (EFA) and Confirmatory Factor Analysis (CFA). Reliability was revealed through Cronbach's alpha coefficient (Cronbach, 1951), and Multidimensional Item Response Theory (MIRT) was conducted to assess the item parameters: item difficulty and item discrimination levels.

3.3.1. Exploratory Factor Analysis (EFA)

Firstly, a Confirmatory Factor Analysis (CFA) was conducted to test the target model of CT, which was constructed based on the components originally intended during the item development process.

However, the initial CFA results did not demonstrate a satisfactory model fit. Therefore, an Exploratory Factor Analysis (EFA) was subsequently carried out to empirically explore the underlying structure of the test and the factor loadings of the items.

EFA was conducted using the psych package in R (Revelle, 2025) employing the oblique rotation technique (Gorsuch, 1983). EFA is explained as an exploration of the underlying constructs of the test (Fabrigar and Wegener, 2012). Thus, the underlying factor structure of the instrument and how effectively the items align with the four essential components of computational thinking: abstraction, algorithm, pattern recognition, and decomposition, were investigated. The oblique rotation technique

was applied because it presumes that the four factors are interrelated (Gorsuch, 1983; Brown, 2009). As the data is dichotomous, the Weighted Least Squares (WLS) estimation method was conducted to fit the model to a matrix of tetrachoric correlation (Jöreskog, 1994; Kubinger, 2003).

Commodities were examined to assess item quality. Items 24, 5, 16, and 7 were removed from the test because they have low communalities (less than .20), indicating poor shared variance (Child, 2006). Also, item 1 was identified as a possible Heywood case, which indicates the factor loading is larger than 1.00 and therefore, this item was also removed from the analysis (Harman and Fukuda, 1966; Wang et al., 2023).

After removing problematic items, the EFA was conducted again with the remaining 21 items with four factors. The second EFA analysis result showed a good model fit after removing the five items.

3.3.2. Confirmatory Factor Analysis (CFA)

After elimination of some items as a result of EFA, Confirmatory Factor Analysis (CFA) was conducted using a lavaan package in R (Rosseel, 2012). CFA focuses on determining how well the hypothesized model fits the collected data (Brown, 2015). Therefore, the factor structure revealed through EFA was tested to see the extent to which the instrument demonstrates construct validity in measuring the distinct components of CT skills (abstraction, algorithm, pattern recognition, and decomposition).

According to Ullman (2006), conducting CFA involves four key steps: model specification, model estimation, model evaluation, and model modification.

In the first step, model specification, the hypothesized model, which is recommended by the EFA, should explicitly be defined by recognizing the latent variables, which represent the theoretical concepts. The hypothesized model is based on the four-component model of CT, which consists of four latent variables: abstraction, algorithm,

pattern recognition, and decomposition.

Secondly, the model estimation step involves the estimation of parameters of the specified model using appropriate statistical techniques, such as Maximum Likelihood (ML), GLS (generalized least squares), and Weighted Least Squares (WLS). While ML is commonly used with continuous data, WLS estimation can be preferred in models that contain ordinal data (Oğuzhan, 2023).

Although WLS can be used for binary data, Diagonally Weighted Least Squares (DWLS) estimation was employed in this study because it is more robust and computationally efficient for models with binary items and small-to-moderate sample sizes (Baghdarnia et al., 2014). DWLS uses only the diagonal of the weight matrix, reducing sensitivity to sample size and distributional assumptions. This provides stable parameter estimates even when the data are non-continuous or slightly skewed.

In this phase, factor loadings and correlations between latent variables were calculated based on the observed data. Accurate estimation is essential to assess how well the model represents the underlying structure of the data.

Thirdly, after determining the hypothesized model, which was structured by EFA in this study, estimating parameters, the hypothesized model was evaluated to determine whether they have a good fit or not, in terms of having the minimum difference between the population and sample covariance matrices. Goodness-of-fit indices such as Root Mean Square Error of Approximation (RMSEA), Comparative Fit Index (CFI), and Tucker Lewis Index (TLI) were used to evaluate model fit (Ullman, 2006; Hu and Bentler, 1999). RMSEA values below .06 indicate a close fit, whereas values above .10 suggest a poor fit (Hu and Bentler, 1999). For CFI, values bigger than .95 indicate a good fit model (Hu and Bentler, 1999). TLI values near .90 or .95 are generally considered indicative of good model fit (Schumacker and Lomax, 2010).

Lastly, model modification is carried out to test the hypothesized model and to enhance model fit (Ullman, 2006). If the calculated fit indices do not meet the desired fit criteria, the modification of the model is recommended to meet the fit criteria (Arpacıoğlu, 2023). There are three methods for model modification: chi-square difference, Lagrange multiplier (LM), and Wald tests (Ullman, 2006). The chi-square method was used for model modification.

3.3.3. Reliability

Internal consistency was assessed for each factor, and the overall test using Cronbach's alpha (Cronbach, 1951).

3.3.4. Item Analysis

To assess item-level performance, Multidimensional Item Response Theory (MIRT) was conducted using the mirt package in R (Chalmers, 2012).

3.3.4.1. Multidimensional Item Response Theory (MIRT). Multidimensional Item Response Theory (MIRT) can be defined as a framework that models the relationship between two or more constructs or dimensions and the probability of a test-taker responding to any question correctly. Unlike Classical Test Theory (CTT), which focuses on observed scores as the sum of true and random error scores, MIRT provides the performance of examinees on item-level parameters of each item across dimensions (Crocker and Algina, 1986). Therefore, the test takers can be differentiated across different CT components through MIRT. Also, the MIRT model can be used to deeply understand the constructs that are wanted to be assessed, as well as the effectiveness of the items that aim to measure these constructs (Ackerman, 1994).

Item Response Theory (IRT) encompasses a family of mathematical models designed to analyze the relationship between an individual's latent ability and their responses to test items. The specific model chosen depends on the characteristics of

the data, such as the item response format (e.g., dichotomous or polychotomous) and the theoretical assumptions about the items (Hambleton and Jones, 1993). For dichotomous items, three common logistic models are used to model the probability of a correct response: the One-Parameter Logistics (1-PL) model, the Two-Parameter Logistics (2-PL) model, and the Three-Parameter Logistics (3-PL) model (Sheng and Wikle, 2007).

The MIRT approach was selected as it directly aligns with the multidimensional structure of the four-component framework. MIRT allows a more comprehensive evaluation of item performance by modeling the influence of multiple latent traits on each item response simultaneously. Within this framework, a two-parameter logistic (2-PL) model was employed to estimate key item parameters: item difficulty and item discrimination.

According to Reckase (2009), although classical IRT assumes unidimensionality, its core assumptions—unidimensionality, monotonicity, and local independence—can be extended for use in multidimensional contexts. Therefore, for MIRT, these assumptions are conceptualized as dimensionality (allowing multiple latent traits), monotonicity, and local independence (Reckase, 2009). The first assumption, dimensionality, refers to the idea that responses to items are influenced by a set of underlying latent traits. It is assumed that the test might assess more than one CT dimension, indicating that each dimension represents separate CT constructs. The second assumption, monotonicity, means that the probability of choosing the correct response increases when the level of the underlying latent trait increases. For instance, it is assumed that a student with a higher ability in abstraction is expected to answer items in the abstraction construct correctly than those with lower abstraction skill. The third assumption of local independence was evaluated, which proposes that a student's response to an item is not influenced by the response to other items when the student's level of latent trait is considered.

The first assumption is that dimensionality is assessed through EFA analysis and parallel analysis to ensure the multidimensional nature of the data (Horn, 1965). The second assumption, monotonicity, was examined through item characteristic curves (ICCs) to determine whether the probability of students' correct responses increases monotonically as the latent trait level increases.

For the assumption of local independence, Yen's Q3 was calculated (Yen, 1984). This assumption was assessed by examining the residual correlations among item pairs after accounting for the latent traits. Specifically, high residual correlations would suggest local dependence, whereas correlations close to zero indicate that item responses are conditionally independent given the multidimensional ability estimates. This evaluation ensured that each item functions independently once the latent traits are taken into account.

Finally, the item difficulty and item discrimination were calculated to evaluate the quality of the items. Item difficulty reflects the level of challenge of an item and is calculated as the proportion of correct responses to the total number of responses. On the other hand, item discrimination provides insights into how well each item differentiates between high- and low-performing students and how strongly each item relates to the constructs.

4. FINDINGS

This chapter presents the results of the psychometric analyses conducted to evaluate the Computational Thinking Skills Test (CTST) following its administration to 429 fifth and sixth-grade students. The findings are organized into three main sections corresponding to the study's research questions. The first section details the evidence for the test's validity, including content validity derived from expert reviews and construct validity from EFA and CFA. The second section reports the internal consistency reliability of the instrument's subscales. The final section presents the results of the item-level analysis, including item difficulty and discrimination parameters, conducted using MIRT.

4.1. Validity

The validity of the instrument was evaluated through two primary forms of evidence: content validity, derived from expert review, and construct validity, which was investigated through factor analysis.

4.1.1. Content Validity of The Final Implementation

To establish the content validity, the initial pool of 41 items was evaluated by a panel of eight experts. The experts evaluated each item's alignment with the four CT constructs, its appropriateness for the target age group, and its perceived difficulty level (easy, medium, or hard).

Each item was rated as appropriate (1) or not appropriate (0). Based on the expert ratings, the Item-Level Content Validity Index (I-CVI) was calculated for each item (see Table 4.1), and the Scale-Level Content Validity Index based on the average method (S-CVI/Ave) was calculated for the overall test (Lynn, 1986; Polit et al., 2007). When eight experts are consulted, the acceptable CVI value is at least 0.80,

recommended by Lynn (1986). The S-CVI/Ave for the overall test is 0.81, indicating a satisfactory value for the appropriateness of the items with the test. Eighteen items were below this threshold. Of these, fifteen were eliminated. Three of them were revised not because of conceptual flaws, but due to issues related to their visual design or clarity of wording.

Table 4.1: Experts I-CVI scores

Items	**The Number of Experts in Agreement**	**I-CVI Score**	**Items**	**The Number of Experts in Agreement**	**I-CVI Score**
7	8	1	38	7	0.875
12	8	1	41	7	0.875
14	8	1	8	6	0.75
21	8	1	10	6	0.75
23	8	1	15	6	0.75
24	8	1	18	6	0.75
33	8	1	20	6	0.75
34	8	1	25	6	0.75
40	8	1	26	6	0.75
1	7	0.875	29	6	0.75
2	7	0.875	30	6	0.75
3	7	0.875	37	6	0.75
4	7	0.875	11	5	0.625
5	7	0.875	17	5	0.625
6	7	0.875	19	5	0.625
9	7	0.875	31	5	0.625
13	7	0.875	35	5	0.625
16	7	0.875	39	5	0.625
27	7	0.875	22	3	0.625
28	7	0.875	36	3	0.625
32	7	0.875			

The difficulty level of each item, as rated by the experts, was scored on a three-point scale: easy (1), medium (2), and hard (3). The mean difficulty score of the remaining 26 items was found to be 1.75, placing the test within the medium difficulty range. Based on this distribution, difficulty bands were defined as easy (1.00–1.65),

medium (1.66–2.32), and hard (2.33–3.00). At the end, the items were arranged in order from easiest to hardest question (see Table 4.2).

Table 4.2: The difficulty average of items

Items	**Difficulty Average**	**Difficulty Level**	**Items**	**Difficulty Average**	**Difficulty Level**
1	2.13	Medium	21	1.38	Easy
2	1.00	Easy	23	1.38	Easy
3	1.50	Easy	24	1.75	Medium
4	1.25	Easy	25	2.50	Hard
5	1.88	Medium	26	2.56	Hard
6	1.50	Easy	27	1.75	Medium
7	2.75	Hard	28	1.71	Medium
9	1.71	Medium	32	1.29	Easy
12	1.13	Easy	33	1.50	Easy
13	1.88	Medium	34	1.38	Easy
14	2.00	Medium	38	1.29	Easy
15	2.43	Hard	40	1.50	Easy
16	2.14	Medium	41	2.31	Medium

4.1.2. Construct Validity

Evidence for construct validity was established through a two-phase analytical approach, employing both Exploratory Factor Analysis (EFA) and Confirmatory Factor Analysis (CFA).

4.1.2.1. Exploratory Factor Analysis (EFA). CFA was initially performed to evaluate the target model of CT, which had been structured according to the components initially targeted during the item development phase (see Table 4.3). CFA was initially performed to evaluate the target model of CT, which had been structured according to the components initially targeted during the item development phase (see Table 4.3).

Table 4.3: Target model of CT

Item Number	Component	Item Number	Component
1	Algorithm	14	Algorithm
2	Pattern Recognition	15	Decomposition
3	Abstraction	16	Pattern Recognition
4	Decomposition	17	Pattern Recognition
5	Abstraction	18	Decomposition
6	Algorithm	19	Abstraction
7	Abstraction	20	Pattern Recognition
8	Abstraction	21	Abstraction
9	Algorithm	22	Decomposition
10	Abstraction	23	Decomposition
11	Pattern Recognition	24	Algorithm
12	Decomposition	25	Pattern Recognition
13	Abstraction	26	Abstraction

However, the initial CFA results did not demonstrate a satisfactory model fit, as correlations between some factors exceeded 1.00, suggesting possible problems with the proposed factor structure.

Therefore, Exploratory Factor Analysis (EFA) was employed to identify the factor structure of the test. As factors expected to be correlated, the Weighted Least Squares (WLS) estimation method was conducted with an oblique rotation technique (Gorsuch, 1983; Jöreskog, 1994). In order to check the suitability of the data for factor analysis, Bartlett's Test of Sphericity (Bartlett, 1954) and the Kaiser-Meyer-Olkin (KMO) measure of sampling adequacy (Kaiser, 1974; Kaiser and Rice, 1974) were conducted. KMO value is found to be 0.66, which is considered mediocre and indicates an adequate value but lower end of adequacy for factor analysis (Kaiser, 1974). To evaluate whether the correlation matrix is significantly varied from an identity matrix, Bartlett's Test of Sphericity was conducted. The result was found to be statistically significant, $^{2}(325) = 753.72$, $p < .001$, suggesting that the items are adequately interrelated and the dataset is appropriate for EFA (Shrestha, 2021).

According to the Kaiser's criterion, ten factors with eigenvalues greater than one were identified (Kaiser, 1974; Kaiser and Rice, 1974). On the other hand, parallel analysis suggested eleven factors (Horn, 1965). However, considering the theoretical basis, the four-component model of CT (abstraction, algorithm, pattern recognition, and decomposition) of the test and the limited number of items (n=26), extracting such a high number of factors would not be appropriate. According to MacCallum et al. (1999), a higher number of factors requires a larger item pool to ensure factor loadings and a reliable representation of each construct. Therefore, the EFA was conducted with the four factors, which were determined according to the four-component model (see Table 4.4).

Table 4.4: The factor loadings and communality values of the initial test.

Item Number	**Factor 1**	**Factor 2**	**Factor 3**	**Factor 4**	**Communality (h^2)**
1	0.86				0.72
2	0.35				0.30
3			0.42		0.24
4				0.36	0.39
5		0.33	-0.30		0.19
6	0.43			0.40	0.38
7	0.41				0.25
8				0.48	0.34
9			0.36		0.31
10	0.44				0.25
11			0.43		0.25
12	0.34				0.20
13				0.45	0.23
14					0.20

Table 4.4 The factor loadings and communality values of the initial test (cont.).

Item Number	**Factor 1**	**Factor 2**	**Factor 3**	**Factor 4**	**Communality (h^2)**
15			0.33		0.18
16			0.60		0.37
17					0.14
18		0.32	0.41		0.34
19		0.34	0.34		0.25
20					0.19
21			0.35		0.17
22		0.33			0.13
23		0.54			0.33
24					0.06
25		0.59			0.36
26				0.55	0.39

Firstly, communalities were examined to assess how much of each item's variance was explained by the common factors in the model (Field, 2009). As a result, four items (24, 5, 16, and 7) were removed from the test due to low communalities (below .20), indicating poor shared variance (Child, 2009). Also, item 1 was excluded from the analysis because it was a possible Heywood case, having a factor loading larger than 1.00 (Harman and Fukuda, 1966; Wang et al., 2023).

Then, EFA was re-conducted with the remaining 21 items with four factors. The second EFA showed improved and clearer factor loadings compared to the initial EFA (see Table 4.5).

Table 4.5: Factor loadings and communality values of the final test

Item Number	**Factor 1**	**Factor 2**	**Factor 3**	**Factor 4**	**Communality (h^2)**
2				0.36	0.29
3	0.32				0.14
4			0.30	0.33	0.34
6			0.30		0.28
8			0.50		0.40
9	0.48				0.34
10	0.31				0.22
11	0.68				0.50
12	0.49				0.24
13			0.51		0.25
14				0.38	0.22
15				0.37	0.23
17				0.41	0.17
18				0.47	0.29
19				0.44	0.25
20		0.30			0.21
21				0.42	0.17
22				0.46	0.21
23		0.30		0.40	0.30
25		0.94			0.90
26			0.50		0.31

Although some items still had relatively low communalities, they were kept in the model to maintain a number of items per factor.

The factors were identified based on the factor loadings of each item. Items 3, 9, 10, 11, and 12 are grouped under Factor 1, while items 20 and 25 were loaded onto Factor 2. Factor 3 was represented by items 6, 8, 13, 26, whereas Factor 4 included items 2, 14, 15, 17, 18, 19, 21, and 22. Item 4 loaded onto both Factor 3 and 4, and item 23 loaded onto both Factors 2 and 4 (see Table 4.5).

After determining the factor loadings each item, the factors were named based on expert opinions, and Model 1 was constructed accordingly (see Table 4.6).

Table 4.6: Model 1

Item Number	**Factors**	**Item Number**	**Factors**
2	Decomposition	15	Decomposition
3	Algorithm	17	Decomposition
4	Decomposition	18	Decomposition
6	Abstraction	19	Decomposition
8	Abstraction	20	Pattern Recognition
9	Algorithm	21	Decomposition
10	Algorithm	22	Decomposition
11	Algorithm	23	Pattern Recognition
12	Algorithm	25	Pattern Recognition
13	Abstraction	26	Abstraction
14	Decomposition		

Item 9 was constructed in order to measure the algorithm construct in line with Model 1. Items 3, 10, and 11 were determined by experts' opinions as an algorithm construct. Therefore, Factor 1 is named Algorithm.

As items 20, 23, and 25 were hypothesized as pattern recognition in Model 1, Factor 2 was named Pattern Recognition.

The items 8, 13, and 26 were generated as an abstraction construct in Model 1. Therefore, Factor 3 was named Abstraction.

As items 4, 15, 18, and 22 were constructed as a decomposition in Model 1. Therefore, Factor 4 was evaluated as decomposition.

However, in Model 1, some items (2, 6, 12, 14, 17, 19, and 21) did not load onto any of the hypothesized components as originally intended. Following EFA, these items were grouped with factors based on their highest factor loadings, and the factor names were assigned based on the other items that were loaded onto the same factor (Hair et al., 2019).

4.1.2.2. Confirmatory Factor Analysis (CFA). Following the EFA, CFA was conducted to test the fit of the proposed four-component structure of the CTST. The analysis was based on the 21-item solution derived from the EFA, which is Model 1.

The Model 1 was evaluated using goodness-of-fit indices, including the Root Mean Square Error of Approximation (RMSEA), the Comparative Fit Index (CFI), and the Tucker-Lewis Index (TLI). The CFA results indicated an acceptable model fit for Model 1, though not an excellent fit. The RMSEA value of 0.021 was below the 0.06 threshold for a close fit. However, the CFI (0.909) and TLI (0.896) values were below the conventional 0.95 threshold for good fit. Examination of the standardized factor loadings for Model 1 revealed that Item 26 had a very low loading of 0.11.

Consequently, a modified model (Model 2) was tested, and Item 26 was removed. As shown in Table 4.7, Model 2 demonstrated an improved fit, with the CFI increasing to 0.933 and the TLI to 0.923, both indicating an acceptable model fit. A chi-square difference test confirmed that removing Item 26 did not significantly worsen the model fit ($\Delta\,^2(1) = 2.41$, p = .12), so the more parsimonious Model 2 was retained as the preferred model (see Table 12).

Table 4.7: The fit indices in CFA

Model	χ^2	**df**	**CFI**	**TLI**	**RMSEA**
Model 1	193.42	183	0.909	0.896	0.021
Model 2	195.83	184	0.933	0.923	0.019

Moreover, the standardized factor loadings of the final 20-item model are presented in Table 4.8. While many items demonstrated satisfactory loadings above the recommended 0.40 threshold, several items fell below this value (minimum 0.30). These items were retained to ensure each of the four factors was represented by at least three items, and a decision was made to preserve the theoretical breadth of the instrument.

Table 4.8: Standardized factor loadings for the Model 2

Item Number	**Factors**	**Factor Name**	**Loading**
2	4	Decomposition	.54
3	1	Algorithm	.38
4	4	Decomposition	.58
6	3	Abstraction	.49
8	3	Abstraction	.58
9	1	Algorithm	.59
10	1	Algorithm	.48
11	1	Algorithm	.61
12	1	Algorithm	.31
13	3	Abstraction	.18
14	4	Decomposition	.43
15	4	Decomposition	.41
17	4	Decomposition	.33
18	4	Decomposition	.56
19	4	Decomposition	.42
20	2	Pattern Recognition	.52
21	4	Decomposition	.38
22	4	Decomposition	.25
23	2	Pattern Recognition	.54
25	2	Pattern Recognition	.56

Finally, the correlations between the latent factors in the EFA model were examined (see Table 4.9). The analysis revealed notably high correlations, particularly between Factor 1 (Algorithm) and Factor 3 (Abstraction) at 0.78, and between Factor 1 (Algorithm) and Factor 4 (Decomposition) at 0.61. These high correlations challenge the discriminant validity of the constructs, suggesting a significant overlap and indicating that they may not be measuring entirely distinct skills as intended by the theoretical framework.

Table 4.9: Correlations between latent factors for the preferred model (Model 2)

	F1	**F2**	**F3**	**F4**
F1 (Algorithm)	1.00			
F2 (Pattern Recognition)	0.35	1.00		
F3 (Abstraction)	0.78	0.08	1.00	
F4 (Decomposition)	0.61	0.48	0.62	1.00

In summary, while the CFA provides evidence for an acceptable four-factor model structure, the analysis also revealed significant weaknesses. The low factor loadings for several retained items and the high inter-factor correlations suggest that the four theoretical constructs are not as empirically distinct as hypothesized.

4.2. Reliability of The Final Implementation

Internal consistency reliability was assessed for the overall 21-item scale and for each of the four subscales using Cronbach's alpha coefficients. The analysis revealed that while the overall scale reliability was questionable, the subscale coefficients were critically low, as detailed in Table 4.10.

The overall reliability for the 20-item test was $\alpha = .61$. While this value approaches the lower limit of acceptability for exploratory research, it remains below the preferred .70 threshold for a developed instrument (Nunnally and Bernstein, 1994). More concerning alpha values for the individual subscales. The alpha values for Abstraction ($\alpha = .30$), Pattern Recognition ($\alpha = .33$), and Algorithms ($\alpha = .42$) were

exceptionally poor. The Decomposition subscale, despite having the largest number of items, yielded the highest coefficient, yet its value ($\alpha = .50$) still falls far below the minimally acceptable threshold of .70 for psychometric research (Nunnally and Bernstein, 1994). These findings suggest that the items within each subscale do not consistently measure a single, coherent construct. Consequently, the CTST subscales, in their current form, cannot be considered reliable measures and require further refinement or item revision to achieve adequate reliability.

Table 4.10: The reliability coefficients for the subscales and the overall scale

Components	**Number of items (N)**	**Cronbach's Alpha (α)**
Abstraction (F3)	4	.30
Algorithm (F1)	5	.42
Pattern Recognition (F2)	3	.33
Decomposition (F4)	9	.50
Overall Scale	20	.61

4.3. Item Analysis

To assess the item-difficulty (b) and item discrimination (α), an item analysis was conducted using Multidimensional Item Response Theory (MIRT), Two Parameter Logistics (2-PL) model.

Prior to estimating the parameters, the three assumptions of MIRT were checked. As no assumption violations were observed, the model estimation proceeded. Then, item difficulty and item discrimination were calculated.

To evaluate the first assumption, dimensionality of the MIRT model, EFA, and the parallel analysis results were evaluated to determine the number of dimensions of the test. The parallel analysis suggested eleven factors. However, the dimensionality for the MIRT model was constrained to four dimensions to align with the study's guiding four-component theoretical framework. This decision acknowledges the discrepancy between the exploratory data and the confirmatory theoretical model being tested.

To provide visual evidence for the monotonicity assumption, the Item Characteristic Curves (ICCs) for all 20 items retained in the final model are presented in Figures 5 through 8, organized by the four latent constructs. Figure 5 displays the three items for the Pattern Recognition factor, and Figure 6 shows the three items loading on the Abstraction factor. Figure 7 displays the ICCs for the nine items loading on the Decomposition factor. Figure 8 presents the curves for the five items measuring the Algorithm factor. As illustrated across all four figures, each item exhibits a monotonically increasing trace, where the probability of a correct response ($P(\theta)$) consistently rises with an increase in student ability (θ). This provides strong visual support that the monotonicity assumption was met for all items across all four dimensions. Furthermore, the varying steepness and position of the curves visually confirm the wide range of item discrimination and difficulty parameters reported in Table 4.11.

The final assumption, local independence, was checked using Yen's Q3 statistic, which assesses the residual correlations between items when the latent abilities were controlled. According to Yen (1984), Q3 values below 0.30 mean the acceptable value for the acceptance of the local independence assumption. The results showed that Q3 residual correlations of the items ranged from -0.18 to 0.19, falling below the suggested threshold. Therefore, there is no item that violates the assumption of local independence.

With the assumptions satisfied, the item parameters were estimated. As shown in Table 4.11, item discrimination (a) values ranged from 0.46 to 2.22, indicating that most items showed moderate to high discrimination and were effective at differentiating between students with varying levels of ability. In Item Response Theory, the discrimination parameter indicates how effectively an item differentiates between students at different levels of the latent trait (Reckase, 2009). Following common interpretation guidelines, discrimination values above 1.70 are considered very high, values from 0.65 to 1.69 are moderate to high, and those from 0.35 to 0.64 are low (Baker, 2001). The range of values in this study indicates that while most items demonstrated acceptable discrimination, a few exhibited low discrimination (e.g., Item 13, a = 0.46), suggesting

they provide limited information for differentiating among students. This variability is visually represented by the steepness of the Item Characteristic Curves presented in Figures 4.1-4.4.

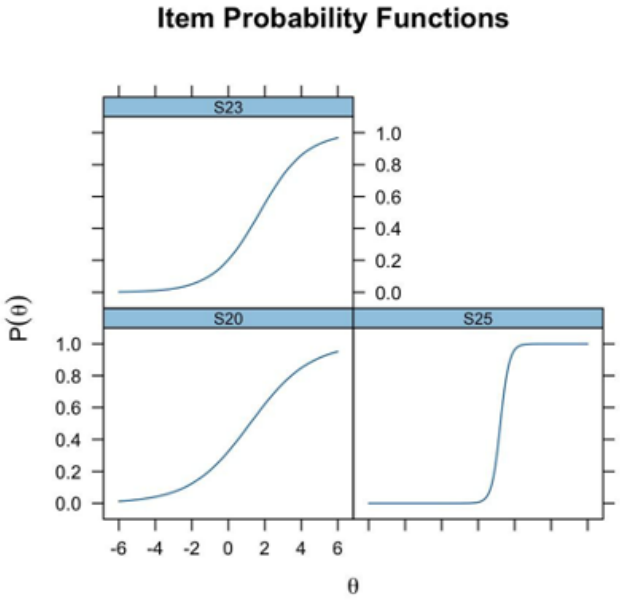


Figure 4.1: ICCs of items for Pattern Recognition.

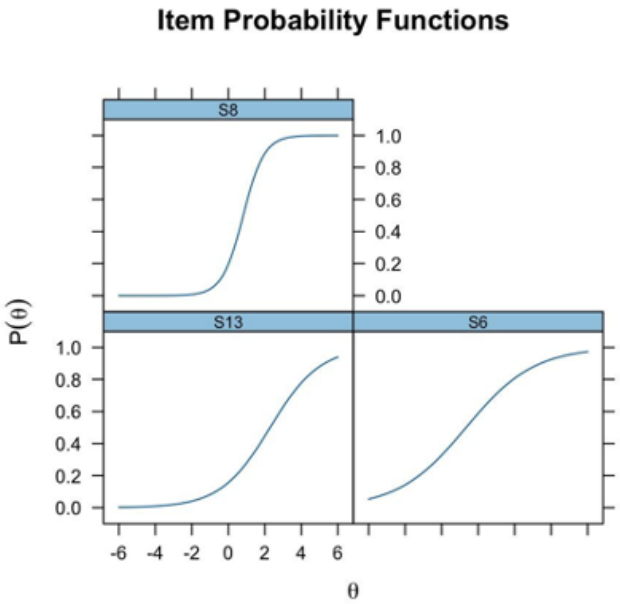


Figure 4.2: ICCs of items for Abstraction.

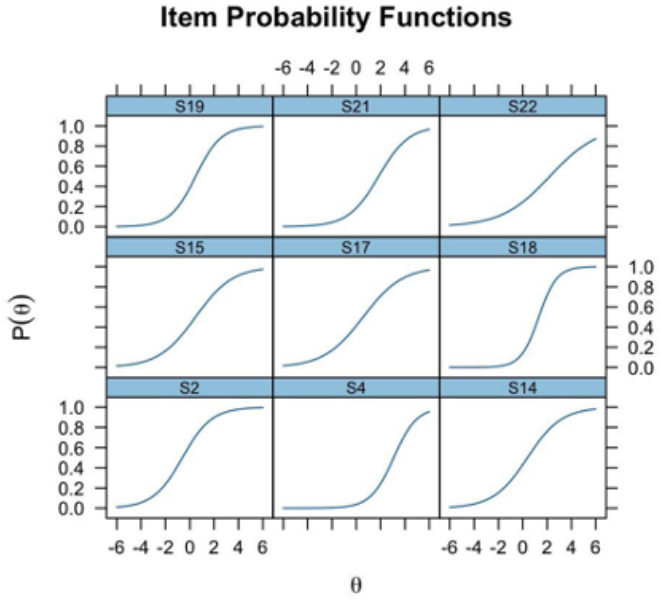


Figure 4.3: ICCs of items for Decomposition.

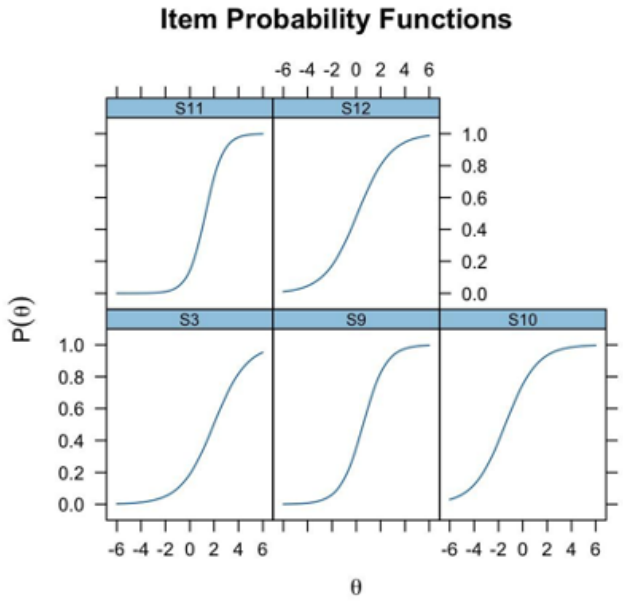


Figure 4.4: ICCs of items for Algorithm.

Item difficulty (b) values ranged from -1.52 to 3.48, demonstrating that the test includes a wide spectrum of items, from easy to very difficult. This broad range of dif-

ficulty and discrimination parameters suggests the instrument is capable of measuring a wide range of student ability levels.

Table 4.11: Item discrimination and difficulty parameters

Item	a_1	a_2	a_3	a_4	b
2	0,84	0,00	0,00	0,00	-0,61
3	0,00	0,72	0,00	0,00	2,01
4	1,06	0,00	0,00	0,00	3,06
6	0,00	0,00	0,00	0,82	-0,47
8	0,00	0,00	0,00	1,20	0,99
9	0,00	1,35	0,00	0,00	0,48
10	0,00	0,71	0,00	0,00	-1,52
11	0,00	1,30	0,00	0,00	1,34
12	0,00	0,54	0,00	0,00	0,14
13	0,00	0,00	0,00	0,46	3,48
14	0,72	0,00	0,00	0,00	0,38
15	0,66	0,00	0,00	0,00	0,47
17	0,62	0,00	0,00	0,00	0,57
18	1,36	0,00	0,00	0,00	1,31
19	0,96	0,00	0,00	0,00	0,48
20	0,00	0,00	0,97	0,00	0,83
21	0,80	0,00	0,00	0,00	1,84
22	0,51	0,00	0,00	0,00	2,22
23	0,00	0,00	0,72	0,00	1,84
25	0,00	0,00	1,90	0,00	1,52

5. DISCUSSION, IMPLICATIONS, LIMITATIONS, AND FUTURE DIRECTIONS

This study embarked on an ambitious and necessary goal: to develop a valid and reliable instrument to assess the computational thinking (CT) skills of 5th and 6th-grade students in a non-programming context. Grounded in the prevalent four-component theoretical framework, abstraction, algorithm, pattern recognition, and decomposition, this research aimed to address a well-established gap in the literature by creating an accessible and psychometrically sound tool. This chapter synthesizes the findings from the validation process, interprets their meaning within the broader context of CT education, and discusses the study's significant contributions, limitations, and implications for the future. The emerging narrative highlights the development of a promising instrument that has provided valuable insights into the assessment of unplugged CT.

5.1. Discussion

The initial phases of instrument development showed considerable success. The process of grounding the test in established CT frameworks, generating a robust initial item pool, and conducting a rigorous content validity review with eight experts established a strong theoretical and qualitative foundation. The resulting Scale-Level Content Validity Index (S-CVI/Ave) of .81 confirms that the test content is relevant and appropriate, meeting established psychometric standards (Lynn, 1986; Polit et al., 2007).

Furthermore, the item-level analysis conducted through MIRT revealed that some of the items have high psychometric quality. The Item Characteristic Curves (ICCs), presented in Figures 5-8, demonstrate that most of the items have positive discrimination parameters and cover a wide range of difficulty, in line with best practices for item analysis (Reckase, 2009; Baker, 2001). As seen in the ICC plots, items, such as

S25 and S11 show steep curves, indicating they are highly effective at discriminating between students of varying abilities, while the spread of items along the ability axis confirms the test's capacity to measure a broad spectrum of student performance. This indicates that the core components of the instrument are psychometrically sound.

When viewed as a single measure of general CT ability, the instrument shows promise. The overall Cronbach's alpha for the 20-item scale was .61. While this is below the conventional .70 threshold for a finalized instrument, it approaches the level of acceptability for exploratory research and suggests that the items, when taken together, are beginning to measure a common underlying trait related to CT (Nunnally and Bernstein, 1994). This indicates that the fundamental parts of the tool are robust.

The most significant contribution of this study emerged not from a straightforward validation, but from the insights the data presented to the theoretical model. Even though the four components of CT are widely mentioned in literature, this study's findings provide strong empirical evidence that these constructs are not as distinct as hypothesized when measured in an unplugged setting with middle-grade students. This discovery was clarified by three key findings.

First, the Exploratory Factor Analysis (EFA) initially suggested a more complex structure of ten or eleven factors, which was quite different from the four-factor model we had hypothesized. The decision to continue with a four-factor model was a theoretically necessary step for this study, but the EFA's initial result clearly showed that the students' response patterns did not fit with the four-component framework. In other words, the data challenged our assumptions, revealing a possible disconnection between how we conceptualized the constructs and how students actually approached the items.

Second, although the Confirmatory Factor Analysis (CFA) supported the overall fit of the four-factor model, it also revealed a high correlation (0.78) between the latent factors of Algorithm and Abstraction. This level of correlation suggests that these two

constructs are not well differentiated—at least not in the minds of the students. This finding aligns with the literature where, in practical problem-solving, abstraction is often inseparable from an algorithm (a sequence of steps) (Wing, 2006; Wing, 2011).

Interestingly, this reflects what is often observed in literature: in real-world problem-solving, abstraction, recognizing the key elements of a problem, and algorithm, figuring out a sequence of steps, tend to happen together (Wing, 2006; Wing, 2011). Although these two skills are treated as distinct in theory, they may not be easily separated in practice.

Finally, and most critically, these structural concerns were confirmed by the reliability analysis. The Cronbach's alpha values for all four subscales were critically low, ranging from .30 to .50, indicating that the items grouped under each component did not measure a single, consistent component (Nunnally and Bernstein, 1994). This is not only a limitation, it is a profound finding. It suggests that the theoretical distinctions between constructs like "abstraction" and "decomposition" may be clear to experts, but students may not approach or experience these skills as separate processes when solving problems assuming that the test items adequately captured the individual CT components.

Taken together, these findings help resolve the seen contradiction between the acceptable model fit found in the CFA and the low reliability of the subscales. While the CFA results suggest that a four-factor structure can work mathematically, the reliability analysis reveals that these four factors are not meaningful or coherent.

Therefore, these findings suggest that the issue may not be limited to item design alone but could also be related to the theoretical framework itself. Although the four-component model of CT—abstraction, algorithm, pattern recognition, and decomposition—is widely recognized in the literature, there is no full consensus on its dimensionality, and different studies have conceptualized CT in varying ways (e.g., Barr and Stephenson, 2011; Brennan and Resnick, 2012; Grover and Pea, 2013). Our

results indicate that the four-component structure may not be distinctly observable among middle school students in non-programming contexts. In other words, students may not experience or demonstrate these components as clearly separable processes when solving problems. This suggests that the high inter-factor correlations and the divergence between the EFA and CFA results may not only reflect measurement challenges but also raise questions about the validity of the theoretical distinctions themselves. Our findings, therefore, imply that the four-component structure of CT may be less differentiated at this developmental stage, pointing to a need for further research not only on test development but also on the theoretical framing of CT.

The main contribution of this thesis, therefore, is not a finished four-component test, but the crucial, data-driven discovery that the four-component model itself requires significant rethinking for unplugged assessment in this age group. This aligns with the broader challenge in the field; the lack of consensus on defining CT components continues to be a major obstacle in developing widely accepted assessment tools (Shute et al., 2017; Tang et al., 2020).

5.2. Implications

This study has significant implications for both researchers and educators in the field of CT.

The key implication for researchers is that the field must move beyond theoretical consensus and develop more precise, empirically validated, and evidence-based definitions for CT components in non-programming contexts. This has been a recurring challenge in the literature (e.g., Grover and Pea, 2013; Shute et al., 2017). This study highlights a critical point: theoretical distinctions between CT components do not necessarily translate into empirically distinct or psychometrically reliable subscales. In other words, theoretical categories don't always lead to clean, separate scales when measured in practice.

For educators, the current 20-item version of the instrument is not suitable for generating separate scores for the four CT components for diagnostic or summative purposes. However, it holds potential as a formative assessment tool in a more holistic sense. The overall score may provide a general sense of a student's CT ability, and the individual items can be used as instructional tools to initiate classroom discussions around diverse problem-solving strategies.

5.3. Limitations and Future Directions

The limitations of this study should be viewed not as failures but as a clear and constructive roadmap to guide subsequent research in the field. The key limitations include:

- Subscale Reliability: The critically low internal consistency coefficients observed across all subscales constitute a fundamental psychometric weakness for the instrument's current use in making any claims about students' distinct CT skills. Therefore, the current version of the instrument is not suitable for making valid inferences regarding students' distinct CT components.
- Construct Validity: The high inter-factor correlations observed in the CFA lead to serious concerns regarding the discriminant validity of the constructs. These findings suggest considerable conceptual overlap among the four CT dimensions.
- Methodological Constraints: The decision to impose a four-factor structure on the EFA, despite the data suggesting other alternative structures, may have introduced an element of confirmatory bias. This choice, although theoretically motivated, may have limited the empirical robustness of the analysis.
- Sampling Limitations: Even though logistically practical, the use of a convenience sample limits the generalizability of the findings to a broader population.

These limitations point to a productive and necessary agenda for future research. The solution is not just adding more items, but building a better theoretical foundation by revisiting and refining the theoretical and operational definitions that underlie the

constructs. Accordingly, future research efforts should:

(i) Conduct Qualitative Studies: To better understand students' cognitive processes during solving the items, future studies should incorporate interviews or think-aloud protocols. These methods can illuminate how and why students may confuse theoretically distinct components, such as abstraction and algorithm.

(ii) Explore Alternative Structural Models: The empirical findings from this study may serve as a valuable baseline for testing alternative measurement models. These could include, for instance, a three-factor model in which closely related constructs are merged (e.g., abstraction and algorithm design), or a bifactor model that includes both general and specific CT factors.

(iii) Begin a New Instrument Development Cycle: A new instrument should be developed by considering insights gained from this study, with careful attention to construct clarity and item specificity.

Taken together, these recommendations aim to contribute to a more robust and theoretically coherent approach to assessing CT in unplugged settings, particularly among middle school students.

6. CONCLUSION

This study successfully progressed the complex process of developing a non-programming-based CT assessment, advancing from theoretical ideas to thorough, multi-phase validation. Though the main aim of developing a valid and reliable four-component tool was not completely achieved, the research process itself yielded a far more valuable outcome. It has produced a promising prototype for a general CT assessment, a set of high-quality individual items, and, most importantly, various data-driven insights that challenge foundational assumptions in the field.

By demonstrating the empirical difficulties of separating the core components of CT in an unplugged context, this thesis makes a significant and sophisticated contribution. It moves the field beyond the theoretical ideal and toward a more grounded, evidence-based roadmap for the next generation of CT assessment research. This work represents a crucial and successful step forward, establishing the foundation for the future research of truly valid and reliable tools to measure this essential 21st-century skill.

REFERENCES

Abdullah, A. H., Othman, M. A., Ismail, N., Abd Rahman, S. N. S., Mokhtar, M., and Zaid, N. M., 2019, "Development of Mobile Application for The Concept of Pattern Recognition in Computational Thinking for Mathematics Subject," in *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, Yogyakarta, Indonesia, pp. 1–9.

Ackerman, T. A., 1994, "Using Multidimensional Item Response Theory to Understand What Items and Tests are Measuring", *Applied Measurement in Education*, vol. 7, no. 4, pp. 255–278.

Aho, A. V., 2012, "Computation and Computational Thinking", *The Computer Journal*, vol. 55, no. 7, pp. 832–835.

Angeli, C., Voogt, J., Fluck, A., Webb, M., Cox, M., Malyn-Smith, J., and Zagami, J., 2016, "A K–6 Computational Thinking Curriculum Framework: Implications for Teacher Knowledge", *Journal of Educational Technology and Society*, vol. 19, no. 3, pp. 47–57.

Arpacıoğlu, B., 2023, *Developing a Scale on Examinees' Attitudes Toward Computerized Adaptive Testing: Applying Technology Acceptance Model*, Master's Thesis, Boğaziçi University, Istanbul, Turkey.

Baghdarnia, M., Soreh, R. F., and Gorji, R., 2014, "The Comparison of Two Methods of Maximum Likelihood (ML) and Diagonally Weighted Least Squares (DWLS) in Testing Construct Validity of Achievement Goals", *Journal of Educational and Management Studies*, vol. 4, no. 1, pp. 22–38.

Baker, F. B., 1965, "An Investigation of the Sampling Distributions of Item Discrimination indices", *Psychometrika*, vol. 30, no. 2, pp. 165–178.

Baker, F. B., 2001, *The Basics of Item Response Theory*, ERIC Clearinghouse on Assessment and Evaluation, College Park, USA.

Barr, V., and 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*, vol. 2, no. 1, pp. 48–54.

Bartlett, M. S., 1950, "Tests of Significance in Factor Analysis", *British Journal of Psychology*, vol. 3, no. 2, pp. 77-85.

Bati, K., 2018, "Computational Thinking Test (CTT) for Middle School Students", *Mediterranean Journal of Educational Research*, vol. 12, no. 23, pp. 89–101.

Bers, M. U., 2012, "Introduction", in *Designing Digital Experiences for Positive Youth Development: From Playpen to Playground*, Oxford University Press, New York, NY, USA, pp. 3–18.

Bers, M. U., Flannery, L., Kazakoff, E. R., and Sullivan, A., 2014, "Computational Thinking and Tinkering: Exploration of an Early Childhood Robotics Curriculum," *Computers and Education*, vol. 72, pp. 145–157.

Bers, M. U., 2019, "Coding as Another Language: A Pedagogical Approach for Teaching Computer Science in Early Childhood," *Journal of Computers in Education*, vol. 6, no. 4, pp. 499–528.

Bers, M. U., 2020, *Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom*, Routledge, New York, NY, USA.

Bers, M. U., Govind, M., and Relkin, E., 2022, "Coding as Another Language: Computational Thinking, Robotics and Literacy in First and Second Grade," in *Computational Thinking in PreK–5: Empirical Evidence for Integration and Future Directions*, pp. 30–38.

Brennan, K., and Resnick, M., 2012, "New Frameworks for Studying and Assessing the Development of Computational Thinking," in *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*, Vancouver, Canada, vol. 1, pp. 1-25.

Brown, J., 2009, "Choosing the Right Number of Components or Factors in PCA and EFA," *JALT Testing and Evaluation SIG Newsletter*, vol. 13, no. 2, pp. 20-25.

Brown, T. A., 2015, *Confirmatory Factor Analysis for Applied Research* (Second edition), The Guilford Press, New York, USA.

Cansu, F. K., and Cansu, S. K., 2019, "An Overview of Computational Thinking," *International Journal of Computer Science Education in Schools*, vol. 3, no. 1, pp. 17–30.

Chalmers, R. P., 2012, "MIRT: A Multidimensional Item Response Theory Package for the R Environment," *Journal of Statistical Software*, vol. 48, pp. 1–29.

Charlton, T., and Luckin, R., 2012, *Time to Reload? Computational Thinking and Computer Science in Schools*, UCL Institute of Education, London, UK.

Chen, G., Shen, J., Barth-Cohen, L., Jiang, S., Huang, X., and Eltoukhy, M., 2017, "Assessing Elementary Students' Computational Thinking in Everyday Reasoning and Robotics Programming," *Computers and Education*, vol. 109, pp. 162–175.

Child, D., 2006, *The Essentials of Factor Analysis* (Third edition), Continuum International Publishing Group, London, UK.

Çoban, E., and Korkmaz, Ö., 2021, "An Alternative Approach for Measuring Computational Thinking: Performance-Based Platform," *Thinking Skills and Creativity*, vol. 42, pp. 1-25.

Corner, S., 2009, "Choosing the Right Type of Rotation in PCA and EFA," *JALT Testing and Evaluation SIG Newsletter*, vol. 13, no. 3, pp. 20–25.

Corradini, I., Lodi, M., and Nardelli, E., 2017, "Conceptions and Misconceptions About Computational Thinking Among Italian Primary School Teachers," in *Proceedings of the 2017 ACM Conference on International Computing Education Research (ICER '17)*, USA, pp. 136–144.

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., Ng, T., Selby, C., and Woollard, J., 2015, *Computational Thinking – A Guide for Teachers*, Computing At School, Birmingham, UK, pp. 1-18.

Cuny, J., Snyder, L., and Wing, J. M., 2010, "Demystifying Computational Thinking for Non-Computer Scientists," Unpublished manuscript.

Crocker, L., and Algina, J., 1986, *Introduction to Classical and Modern Test Theory*, Holt, Rinehart and Winston, Orlando, FL, USA.

Crocker, L. M., and Algina, J., 2008, *Introduction to Classical and Modern Test Theory*, Cengage Learning, Boston, MA, USA.

Cronbach, L. J., 1951, "Coefficient Alpha and the Internal Structure of Tests," *Psychometrika*, vol. 16, no. 3, pp. 297–334.

Dagienė, V., and Futschek, G., 2008, "Bebras International Contest on Informatics and Computer Literacy: Criteria for Good Tasks," in *Informatics Education—Supporting Computational Thinking: Third International Conference on Informatics in Secondary Schools—Evolution and Perspectives, ISSEP 2008*, Toruń, Poland, pp. 19–30.

Doleck, T., Bazelais, P., Lemay, D. J., Saxena, A., and Basnet, R. B., 2017, "Algorithmic Thinking, Cooperativity, Creativity, Critical Thinking, and Problem Solving: Exploring the Relationship Between Computational Thinking Skills and Academic Performance," *Journal of Computers in Education*, vol. 4, no. 4, pp. 355–369.

Durak, H. Y., and Saritepeci, M., 2018, "Analysis of the Relation Between Computational Thinking Skills and Various Variables With the Structural Equation Model," *Computers and Education*, vol. 116, pp. 191–202.

Fabrigar, L. R., and Wegener, D. T., 2012, *Exploratory Factor Analysis*, Oxford University Press, New York, USA.

Fagerlund, J., Häkkinen, P., Vesisenaho, M., and Viiri, J., 2020, "Assessing 4th Grade Students' Computational Thinking Through Scratch Programming Projects," *Informatics in Education*, vol. 19, no. 4, pp. 611–640.

Field, A., 2009, *Discovering Statistics Using SPSS* (Third edition), Sage Publications, CA, USA.

Gorsuch, R. L., 1983, *Factor Analysis* (Second edition), Psychology Press, New York, USA.

Grover, S., and Pea, R., 2013, "Computational Thinking in K–12: A Review of the State of the Field," *Educational Researcher*, vol. 42, no. 1, pp. 38–43.

Grover, S., 2017, "Assessing Algorithmic and Computational Thinking in K–12: Lessons From a Middle School Classroom," in *Emerging Research, Practice, and Policy on Computational Thinking*, New York, USA: Springer, pp. 269–288.

Grover, S., and Pea, R., 2018, "Computational Thinking: A Competency Whose Time Has Come," in Sentance, S., Barendsen, E., and Schulte, C. (Eds.), *Computer Science Education: Perspectives on Teaching and Learning in School*, Bloomsbury Academic, London, UK, pp. 19–38.

Hair, J. F., Anderson, R. E., Tatham, R. L., and Black, W. C., 1995, *Multivariate Data Analysis*, Prentice-Hall, Englewood Cliffs, NJ, USA.

Hair, J. F., Babin, B. J., Anderson, R. E., and Black, W. C., 2019, *Multivariate Data Analysis* (Eighth edition), Pearson, Harlow, UK.

Haladyna, T. M., Downing, S. M., and Rodriguez, M. C., 2002, "A Review of Multiple-Choice Item-Writing Guidelines for Classroom Assessment," *Applied Measurement in Education*, vol. 15, no. 3, pp. 309–333.

Hambleton, R. K., and Jones, R. W., 1993, "Comparison of Classical Test Theory and Item Response Theory and Their Applications to Test Development," *Educational Measurement: Issues and Practice*, vol. 12, no. 3, pp. 38–47.

Harman, H. H., and Fukuda, Y., 1966, "Resolution of the Heywood Case in the MINRES Solution," *Psychometrika*, vol. 31, no. 4, pp. 563–571.

Herro, D., Quigley, C., Plank, H., Abimbade, O., and Owens, A., 2022, "Instructional Practices Promoting Computational Thinking in STEAM Elementary Classrooms," *Journal of Digital Learning in Teacher Education*, vol. 38, no. 4, pp. 158–172.

Holmbeck, G. N., and Devine, K. A., 2009, "An Author's Checklist for Measure Development and Validation Manuscripts," *Journal of Pediatric Psychology*, vol. 34, no. 7, pp. 691–696.

Horn, J. L., 1965, "A Rationale and Test for the Number of Factors in Factor Analysis," *Psychometrika*, vol. 30, no. 2, pp. 179–185.

Hsu, T. C., Chang, S. C., and Hung, Y. T., 2018, "How to Learn and How to Teach Computational Thinking: Suggestions Based on a Review of the Literature," *Computers and Education*, vol. 126, no. 3, pp. 296–310.

Hu, L. T., and Bentler, P. M., 1999, "Cutoff Criteria for Fit Indexes in Covariance Structure Analysis: Conventional Criteria Versus New Alternatives," *Structural Equation Modeling: A Multidisciplinary Journal*, vol. 6, no. 1, pp. 1–55.

International Society for Technology in Education (ISTE), and Computer Science Teachers Association (CSTA), 2011, *Operational Definition of Computational Thinking for K–12 Education*, ISTE, Washington, DC, USA.

Kaiser, H. F., 1974, "An Index of Factorial Simplicity," *Psychometrika*, vol. 39, no. 1, pp. 31–36.

Kaiser, H. F., and Rice, J., 1974, "Little Jiffy, Mark IV," *Educational and Psychological Measurement*, vol. 34, no. 1, pp. 111–117.

Kang, E. J. S., Donovan, C., and McCarthy, M. J., 2018, "Exploring Elementary Teachers' Pedagogical Content Knowledge and Confidence in Implementing the NGSS Science and Engineering Practices," *Journal of Science Teacher Education*, vol. 29, no. 1, pp. 9–29.

Kalelioglu, F., Gülbahar, Y., and Kukul, V., 2016, "A Framework for Computational Thinking Based on a Systematic Research Review," *Baltic Journal of Modern Computing*, vol. 4, no. 3, pp. 583–596.

Korkmaz, Ö., Çakir, R., and Özden, M. Y., 2017, "A Validity and Reliability Study of the Computational Thinking Scales (CTS)," *Computers in Human Behavior*, vol. 72, pp. 558–569.

Kubinger, K. D., 2003, "On Artificial Results Due to Using Factor Analysis for Dichotomous Variables," *Psychology Science*, vol. 45, no. 1, pp. 106–110.

Jöreskog, K. G., 1994, "Structural Equation Modeling With Ordinal Variables," *Lecture Notes—Monograph Series*, vol. 24, pp. 297–310.

Lai, R. P., 2021, "Beyond Programming: A Computer-Based Assessment of Computational Thinking Competency," *ACM Transactions on Computing Education (TOCE)*, vol. 22, no. 2, pp. 1–27.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., and Werner, L., 2011, "Computational Thinking for Youth in Practice," *ACM Inroads*, vol. 2, no. 1, pp. 32–37.

Li, Y., Xu, S., and Liu, J., 2021, "Development and Validation of Computational Thinking Assessment of Chinese Elementary School Students," *Journal of Pacific Rim Psychology*, vol. 15, no.3, pp. 1-22.

Lynn, M. R., 1986, "Determination and Quantification of Content Validity," *Nursing Research*, vol. 35, no. 6, pp. 382–386.

MacCallum, R. C., Widaman, K. F., Zhang, S., and Hong, S., 1999, "Sample Size in Factor Analysis," *Psychological Methods*, vol. 4, no. 1, pp. 84–99.

Mindetbay, Y., Bokhove, C., and Woollard, J., 2019, "The Measurement of Computational Thinking Performance Using Multiple-Choice Questions," *Proceedings of International Conference on Computational Thinking Education*, pp. 176-179.

Ministry of National Education (MoNE). (2023). *Information Technologies and Software Course Curriculum*. `https://mufredat.meb.gov.tr/ProgramDetay.aspx?PID=1663`, accessed in June 2025.

Moreno-León, J., and Robles, G., 2015, "Dr. Scratch: A Web Tool to Automatically Evaluate Scratch Projects," in *Proceedings of the Workshop in Primary and Secondary Computing Education*, London, UK, pp. 132–133.

National Research Council, 2010, *Report of a Workshop on the Scope and Nature of Computational Thinking*, National Academies Press, Washington, USA, pp. 1-102.

Nunnally, J. C., 1967, *Psychometric Theory*, McGraw-Hill, New York, USA.

Nunnally, J. C., and Bernstein, I. H., 1994, *Psychometric Theory* (Third edition), McGraw-Hill, New York, USA.

Ocak, C., Yadav, A., and Macann, V., 2023, "Using Computational Thinking as a Metacognitive Tool in the Context of Plugged vs. Unplugged Computational Activities," in *Proceedings of the 17th International Conference of the Learning Sciences (ICLS 2023)*, Montréal, Canada, pp. 545–552.

Oğuzhan, E. E., 2023, *A Scale Development for Social and Emotional Learning Skills of Middle School Children*, Master's Thesis, Boğaziçi University, Istanbul, Turkey.

Otterborn, A., Schönborn, K. J., and Hultén, M., 2019, "Investigating Preschool Educators' Implementation of Computer Programming in Their Teaching Practice," *Early Childhood Education Journal*, vol. 48, no. 3, pp. 253–262.

Papert, S., 1980, *Mindstorms: Children, Computers, and Powerful Ideas*, Basic Books, New York, USA.

Peel, A., Sadler, T. D., and Friedrichsen, P., 2022, "Algorithmic Explanations: An Unplugged Instructional Approach to Integrate Science and Computational Thinking," *Journal of Science Education and Technology*, vol. 31, no. 4, pp. 428–441.

Polit, D. F., Beck, C. T., and Owen, S. V., 2007, "Is the CVI an Acceptable Indicator of Content Validity? Appraisal and Recommendations," *Research in Nursing and Health*, vol. 30, no. 4, pp. 459–467.

Reckase, M. D., 2009, *Multidimensional Item Response Theory*, Springer, New York, USA.

Relkin, E., De Ruiter, L., and Bers, M. U., 2020, "TechCheck: Development and Validation of an Unplugged Assessment of Computational Thinking in Early Childhood Education," *Journal of Science Education and Technology*, vol. 29, no. 4, pp. 482–498.

Relkin, E., De Ruiter, L., and Bers, M., 2021, "Learning to Code and the Acquisition of Computational Thinking by Young Children," *Computers and Education*, vol. 169, pp. 1-15.

Relkin, E., Johnson, S. K., and Bers, M. U., 2023, "A Normative Analysis of the TechCheck Computational Thinking Assessment," *Educational Technology and Society*, vol. 26, no. 2, pp. 118–130.

Repenning, A., Basawapatna, A., and Escherle, N., 2016, "Computational Thinking Tools," in *2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*, Cambridge, UK, pp. 218–222.

Revelle, W., 2025, *psych: Procedures for Psychological, Psychometric, and Personality Research* (Version 2.5.6) [R package], Northwestern University, Evanston, Illinois. `https://CRAN.R-project.org/package=psych`, accessed in May 2025.

Rich, K. M., Yadav, A., and Schwarz, C. V., 2019, "Computational Thinking, Mathematics, and Science: Elementary Teachers' Perspectives on Integration," *Journal of Technology and Teacher Education*, vol. 27, no. 2, pp. 165–205.

Román-González, M., 2015, "Computational Thinking Test: Design Guidelines and Content Validation," in *EDULEARN15 Proceedings*, Barcelona, Spain, pp. 2436–2444.

Román-González, M., Moreno-León, J., and Robles, G., 2019, "Combining Assessment Tools for a Comprehensive Evaluation of Computational Thinking Interventions," in *Computational Thinking Education*, Springer, Singapore, pp. 79–98.

Román-González, M., Pérez-González, J. C., and Jiménez-Fernández, C., 2017, "Which Cognitive Abilities Underlie Computational Thinking? Criterion Validity of the Computational Thinking Test," *Computers in Human Behavior*, vol. 72, pp. 678–691.

Rosseel, Y., 2012, "lavaan: An R Package for Structural Equation Modeling," *Journal of Statistical Software*, vol. 48, no. 1, pp. 1–36.

Selby, C., and Woolard, C., 2014, *Computational Thinking: The Developing Definition*, Southampton, UK.

Schumacker, R. E., and Lomax, R. G., 2010, *A Beginner's Guide to Structural Equation Modeling* (Third edition), Routledge/Taylor and Francis Group, New York, USA.

Shen, L., Mirakhur, Z., and LaCour, S., 2024, "Investigating the Psychometric Features of a Locally Designed Computational Thinking Assessment for Elementary Students," *Computer Science Education*, vol. 35, no. 2, pp. 414–433.

Sheng, Y., and Wikle, C. K., 2007, "Comparing Multiunidimensional and Unidimensional Item Response Theory Models," *Educational and Psychological Measurement*, vol. 67, no. 6, pp. 899–919.

Shute, V. J., Sun, C., and Asbell-Clarke, J., 2017, "Demystifying Computational Thinking," *Educational Research Review*, vol. 22, no. 1, pp. 142–158.

Shrestha, N., 2021, "Factor Analysis as a Tool for Survey Analysis," *American Journal of Applied Mathematics and Statistics*, vol. 9, no. 1, pp. 4–11.

Strawhacker, A., and Bers, M. U., 2018, "What They Learn When They Learn Coding: Investigating Cognitive Domains and Computer Programming Knowledge in Young Children," *Educational Technology Research and Development*, vol. 67, no. 3, pp. 541–575.

Tang, X., Yin, Y., Lin, Q., Hadad, R., and Zhai, X., 2020, "Assessing Computational Thinking: A Systematic Review of Empirical Studies," *Computers and Education*, vol. 148, pp. 1-22, 103798.

Tosun, D., 2019, *Development of a Scale on Primary School Teachers' Knowledge and Perception of Dyslexia / İlkokul Öğretmenlerinin Disleksi Bilgisi ve Algısı Üzerine Ölçek Geliştirme*, Master's Thesis, Boğaziçi University, Istanbul, Turkey.

Ullman, J. B., 2006, "Structural Equation Modeling: Reviewing the Basics and Moving Forward," *Journal of Personality Assessment*, vol. 87, no. 1, pp. 35–50.

Umutlu, D., 2022, "An Exploratory Study of Pre-Service Teachers' Computational Thinking and Programming Skills," *Journal of Research on Technology in Education*, vol. 54, no. 5, pp. 754–768.

Uzunboylu, H., Kınık, E., and Kanbul, S., 2017, "An Analysis of Countries Which Have Integrated Coding Into Their Curricula and the Content Analysis of Academic Studies on Coding Training in Turkey," *TEM Journal*, vol. 6, no. 4, pp. 783–791.

Voogt, J., Fisser, P., Good, J., Mishra, P., and Yadav, A., 2015, "Computational Thinking in Compulsory Education: Towards an Agenda for Research and Practice," *Education and Information Technologies*, vol. 20, no. 4, pp. 715–728.

Wang, S., De Boeck, P., and Yotebieng, M., 2023, "Heywood Cases in Unidimensional Factor Models and Item Response Models for Binary Data," *Applied Psychological Measurement*, vol. 47, no. 2, pp. 141–154.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., and Wilensky, U., 2016, "Defining Computational Thinking for Mathematics and Science Classrooms," *Journal of Science Education and Technology*, vol. 25, no. 1, pp. 127–147.

Werner, L., Denner, J., Campe, S., and Kawamoto, D. C., 2012, "The Fairy Performance Assessment: Measuring Computational Thinking in Middle School," in *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, Raleigh, USA, pp. 215–220.

Wing, J. M., 2006, "Computational Thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33–35.

Wing, J. M., 2008, "Computational Thinking and Thinking About Computing," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 366, no. 1881, pp. 3717–3725.

Wing, J. M., 2011, "Research Notebook: Computational Thinking—What and Why," *The Link Magazine*, vol. 6, pp. 20–23.

Wu, S.-Y., and Su, Y.-S., 2021, "Visual Programming Environments and Computational Thinking Performance of Fifth- and Sixth-Grade Students," *Journal of Educational Computing Research*, vol. 59, no. 6, pp. 1075–1092.

Yadav, A., Gretter, S., Hambrusch, S., and Sands, P., 2016, "Expanding Computer Science Education in Schools: Understanding Teacher Experiences and Challenges," *Computer Science Education*, vol. 26, no. 4, pp. 235–254.

Yen, W. M., 1984, "Effects of Local Item Dependence on the Fit and Equating Performance of the Three-Parameter Logistic Model," *Applied Psychological Measurement*, vol. 8, no. 2, pp. 125–145.

Zapata-Cáceres, M., Marcelino, P., El-Hamamsy, L., and Martín-Barroso, E., 2024, "A Bebras Computational Thinking (ABC-Thinking) Program for Primary School: Evaluation Using the Competent Computational Thinking Test," *Education and Information Technologies*, vol. 29, no.12, pp. 14969 - 14998.

APPENDIX A: EXAMPLES of CT ASSESSMENT TOOLS

This appendix presents examples of Computational Thinking (CT) assessment tools reported in the literature.

Table A.1: Examples OF Computational Thinking Assessment Tools

Assessment Name	**Measured Component(s)**	**Age Group**	**Type / Number of Items**	**Accessibility**	**Notes**	**Reference**
Fairy Assessment	Algorithmic thinking, abstraction, modeling	Middle school	Simple Alice content knowledge (8 MCQs)	No	Programming based	Werner et al., 2012
Computational Thinking Test (CT-test 2.0)	decomposition, pattern recognition, abstraction and algorithmic design (directions, loops, conditionals, functions)	10–15 years	28 MCQs	Yes	Block-based programming	Román-González, 2017

Table A.1 Examples of Computational Thinking Assessment Tools (cont.)

Assessment Name	**Measured Component(s)**	**Age Group**	**Type / Number of Items**	**Accessibility**	**Notes**	**Reference**
Computational Thinking Challenge (CTC)	Abstraction, algorithmic thinking, problem-decomposition, generalization, debugging	Upper secondary	Parsons problems, MCQs, tasks	Yes	Computer-based	Lai, 2021
CT Assessment Instrument	Everyday scenarios (13 items) and robotics programming (10 items)	6th grade	15 MCQs + 8 open-ended	Partially	Components are different	Chen et al., 2017
Computational Thinking Performance Test	Logical thinking, generalisation, abstraction	13–14 years	50 MCQs	Partially	Different components; age group is old	Mindetbay et al., 2019

Table A.1 Examples of Computational Thinking Assessment Tools (cont.)

Assessment Name	**Measured Component(s)**	**Age Group**	**Type / Number of Items**	**Accessibility**	**Notes**	**Reference**
Bebras Tasks	abstraction, algorithmic thinking, decomposition, evaluation, generalization	11–16 years and secondary	15–21 MCQs	Partially	Components are intertwined; limited classroom implementation	Dagiene and Futschek, 2008
TechCheck	algorithms, modularity, control structures, representation, hardware/software, debugging	5–9 years	15 MCQs	Partially	Younger age group; different components	Relkin et al., 2020

Table A.1 Examples of Computational Thinking Assessment Tools (cont.)

Assessment Name	**Measured Component(s)**	**Age Group**	**Type / Number of Items**	**Accessibility**	**Notes**	**Reference**
Computational Thinking Assessment (CTA-CES)	abstraction, algorithmic thinking, decomposition, evaluation, pattern recognition	9–12 years	25 MCQs	Partially	Couldn't reach author	Li et al., 2021
The Computational Assessment	decomposition, pattern recognition, abstraction, algorithm	Higher-grade elementary	20 MCQs	Partially	Couldn't reach author	Wu and Su, 2021

APPENDIX B: CT COMPONENTS ACROSS FRAMEWORKS

This appendix summarizes CT components across frameworks in the literature.

Table B.1: CT components across frameworks (Part 1)

CT Component	**Barr and Stephenson (2011)**	**Brennan and Resnick (2012)**	**Grover and Pea (2013)**	**ISTE and CSTA (2011)**	**Angeli et al. (2016)**
Abstraction	✓	✓	✓	✓	✓
Decomposition	✓		✓	✓	✓
Pattern Recognition / Generalization			✓		✓
Algorithm	✓		✓	✓	✓
Automation	✓		✓	✓	
Evaluation			✓		
Parallelization	✓			✓	
Testing & Debugging		✓	✓		
Data Collection	✓				
Data Analysis	✓				
Data Representation	✓				
Simulation	✓				
Incremental & Iterative		✓	✓		
Expressing, Connecting &Questioning		✓			
Reusing & Remixing		✓			
Collaboration & Creativity		✓			
Logic & Logical Thinking			✓		
Conceptualising					
Mathematical Reasoning					

Note. ✓ indicates that the computational thinking (CT) component is explicitly included or emphasized in the corresponding framework or publication.

Table B.2: CT components across frameworks (Part 2)

CT Component	**Repenning et al. (2016)**	**Kalelioğlu et al. (2016)**	**Csizmadia (2015)**	**Wing (2006; 2011)**	**Selby and Woollard (2013)**
Abstraction	✓	✓	✓	✓	✓
Decomposition		✓	✓	✓	✓
Pattern Recognition / Generalization		✓	✓	✓	✓
Algorithm		✓	✓	✓	✓
Automation	✓	✓		✓	
Evaluation		✓	✓	✓	
Parallelization	✓		✓		
Testing & Debugging	✓				
Data Collection					
Data Analysis					
Data Representation					
Simulation					
Incremental & Iterative					
Expressing, Connecting & Questioning					
Reusing & Remixing					
Collaboration & Creativity					
Logic & Logical Thinking					
Conceptualising					✓
Mathematical Reasoning					✓

Note. ✓ indicates that the computational thinking (CT) component is explicitly included or emphasized in the corresponding framework or publication.

APPENDIX C: COMPUTATIONAL THINKING SKILL TEST (CTST)

This appendix includes the Computational Thinking Skill Test (CTST) questions.

Ad-Soyad: **Sınıf:** **Okul No:**

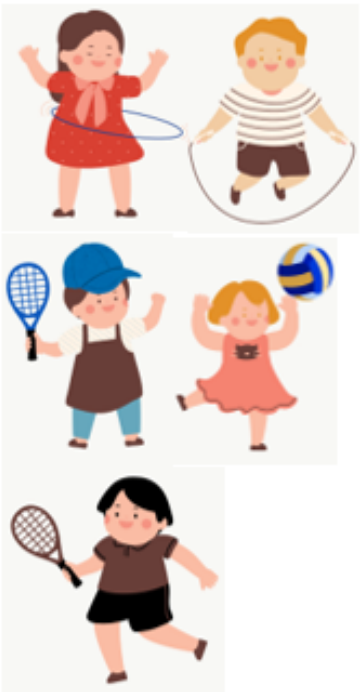
1) Ali ellerini yıkamak için aşağıdaki adımları takip edecektir. Ancak adımların sıralamasında hata yapılmıştır.

1. Musluğu aç ve ellerini ıslat.
2. Ellerine sabun al.
3. Temiz bir havlu veya kağıt havlu ile ellerini kurula.
4. Elleri bol su ile durula.
5. Ellerini en az 20 saniye boyunca ovarak yıka.
6. Musluğu bir havlu veya dirseğinle kapat.

Yukarıda verilen adımlardan hangi ikisi yer değiştirirse, Ali ellerini doğru şekilde yıkar?

A) 3 - 5
B) 1 - 4
C) 2 - 4
D) 4 - 5

2) Zeynep, parkta kaybolan bir çocuğu aramaktadır. Çocuğu bulabilmek için parktaki 5 farklı kişiden çocuğun resmini çizmelerini istemiştir.



Zeynep'in, çizilen 5 farklı resme göre kaybolan çocuk hakkında yaptığı yorumlardan hangisi kesinlikle doğrudur?

A) Kahverengi bir kıyafeti vardır.
B) Elinde tenis raketi vardır.
C) Kahverengi ayakkabı giymektedir.
D) Sarı saçlıdır.

Figure C.1: CTST questions – page 1

3) Aynalı sazan balığı, tatlı sularda yaşayan bir balık türü olmasına rağmen düşük tuzluluk seviyelerine uyum sağlayabilir. Aşağıdaki tabloda balığın yüzebileceği göllerin isimleri, derinlikleri, tuzlulukları ve su sıcaklıkları verilmiştir.

İsim	*Derinlik*	*Tuzluluk*	*Sıcaklık*
Köyiçi Gölü	8 m	997,9 kg/m³	16°C
Ferah Gölü	12 m	1000,13 kg/m³	24°C
Eymir Gölü	7 m	998,9 kg/m³	18°C
Pazar Gölü	4 m	999,7 kg/m³	22°C
Çamlı Gölü	22 m	1000,3 kg/m³	19°C

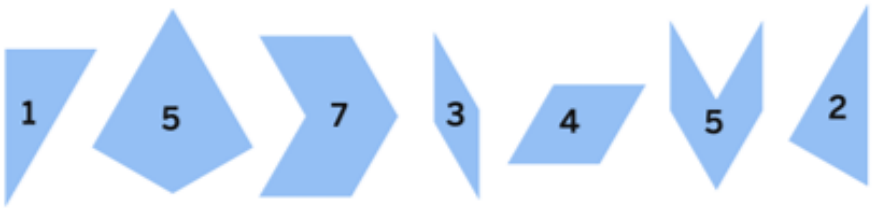
Aynalı sazan balığı, en tuzlu gölden başlayarak en az tuzlu göle kadar yüzmek istiyor. Bu durumda, balığın yüzmesi gereken rota nasıl olmalıdır?

A) Ferah Gölü, Çamlı Gölü, Pazar Gölü, Eymir Gölü, Köyiçi Gölü
B) Ferah Gölü, Çamlı Gölü, Köyiçi Gölü, Eymir Gölü, Pazar Gölü
C) Çamlı Gölü, Pazar Gölü, Eymir Gölü, Köyiçi Gölü, Ferah Gölü
D) Çamlı Gölü, Ferah Gölü, Pazar Gölü, Eymir Gölü, Köyiçi Gölü

4)

Bir müşterisi Terzi Selma Hanım'a aşağıdaki görseldeki kelebek resmini getirerek aynısını dikmesini istemiştir.

Selma Hanım tasarımı oluşturabilmek için aşağıdaki çeşitli çokgen parçalarını kumaştan keserek parçaları hazırlayacaktır. Parçaları hazırlarken şekilleri istediği şekilde döndürebilecek ve kumaşı boyayacak, ardından parçaları birbirine monte edecektir. Her bir çokgen parçasını hazırlamasının kaç dakika sürdüğü çokgen parçalarının üzerine yazılmıştır.



Buna göre, Selma Hanım'ın tasarımı oluşturabileceği minimum süre kaç dakikadır?

A) 24 B) 25 C) 26 D) 27

Figure C.2: CTST questions – page 2

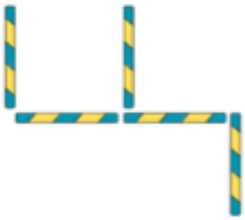
5)

Züleyha, elindeki 5 pipeti kullanarak yandaki şekli oluşturmuştur. Belgin, Züleyha'nın oluşturduğu şekilde bir pipeti hareket ettirerek yeni bir şekil oluşturmuştur. **Buna göre aşağıdakilerden hangisi Belgin'in oluşturduğu şekillerden biri olamaz?**

A)

B)

C)



D)

Figure C.3: CTST questions – page 3

6) Bir büfeci, sandviçleri aşağıdaki kurala göre yapıyor.

1. Domates ekmekle doğrudan temas etmemelidir.
2. Peynir domatesin altında olmalıdır.
3. Salatalık ve peynir marulun üstünde olmalıdır.
4. Tüm malzemeler sandviç ekmeği arasında olmalıdır.

Aşağıda üstten görünümü verilen sandviçlerden hangisi, yukarıdaki kurallara uygun olarak yapılmıştır?

A)

B)

C)

D)

Figure C.4: CTST questions – page 4

7) Üç A4 kâğıdı birden fazla renk kullanılarak farklı desenlerde boyanıyor. Ardından her kâğıttan yapboz parçası şeklinde birer parça kesiliyor.

Kesim işlemi sonucunda yukarıdaki 3 yapboz parçası ortaya çıktığına göre, aşağıdakilerden hangisi yapboz parçalarının kesildiği A4 sayfalarından birisi olamaz?

A)

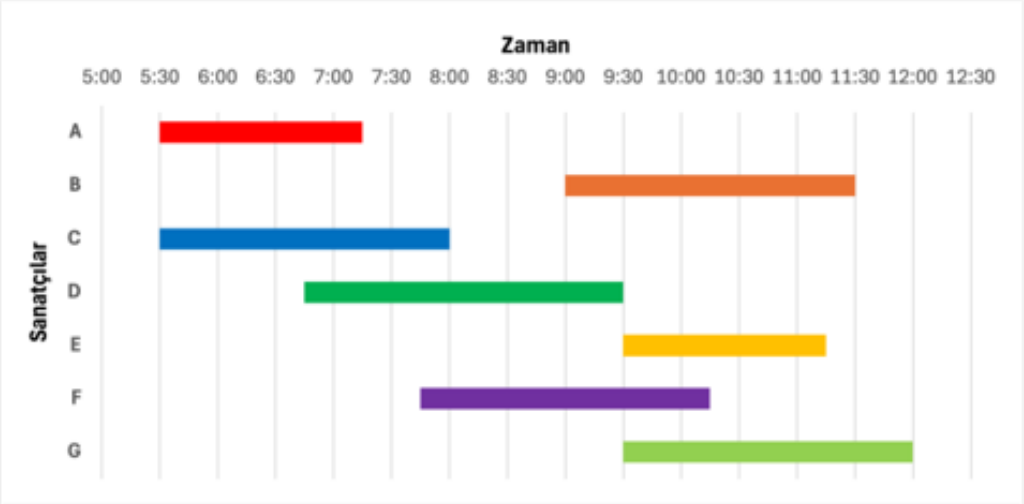
B)

C)

D)

Figure C.5: CTST questions – page 5

8) Ali, bir saat boyunca kalabileceği bir sanat sergisine gidecek ve sergide eserleri bulunan sanatçıların söyleşilerine katılacaktır. Aşağıdaki zaman çizelgesinde, sanatçıların söyleşi programı verilmiştir.



Ali, bir saat içinde en fazla sayıda söyleşiye katılmak istiyorsa, hangi zaman dilimi için bilet almalıdır?

A) 5:30 – 6:30 B) 7:00 – 8:00 C) 8:30 – 9:30 D) 10:30 – 11:30

9) Kütüphane görevlisi Orhan, kitapları belirli kurallara göre sıralı bir şekilde işlemektedir. Her kitap, kategorisinin belirlenmesi, kaydedilmesi ve rafa yerleştirilmesi adımlarından geçerek toplam **30 dakikada** işlenir. Orhan, kitapları geldiği sırayla işleme alır ve bir kitabın işlemleri tamamlanmadan diğerine başlamaz.

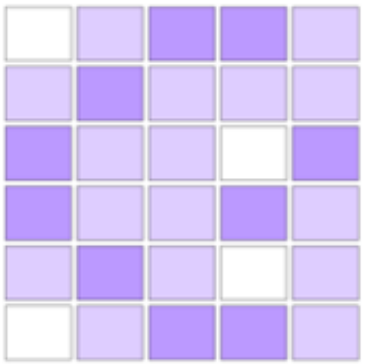
Kitap Türü	**Geliş Saati**
Masal	14.00
Bilimsel- Makale	14.15
Roman	14.30

Orhan, yukarıdaki kurallara ve geliş saatlerine göre kitapları işler. Buna göre, aşağıdaki ifadelerden hangisi yanlıştır?

A) Masal kitabı işlemleri saat 14.30'da tamamlanır.
B) Bilimsel Makale işlemleri saat 15.00'te tamamlanır.
C) Roman işlemleri saat 15.10'da tamamlanır.
D) Tüm kitapların işlemleri saat 15.30'da tamamlanır.

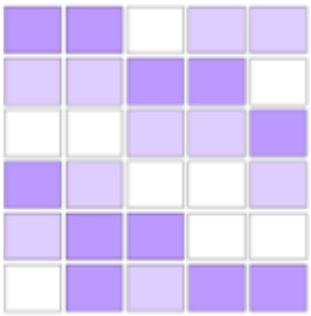
Figure C.6: CTST questions – page 6

10) 6x5 büyüklüğündeki karelerden oluşan bir led ışıklı panoda üç farklı renkte ışık yanmaktadır. Bu renklerin yanma durumuna karşılık gelen rakamlarla, her satıra karşılık gelecek şekilde yukarıdan aşağıya doğru bir şifreleme sistemi oluşturulmuştur. Örneğin, aşağıdaki görseldeki panoya denk gelen şifreleme sistemi yanında verilmiştir.



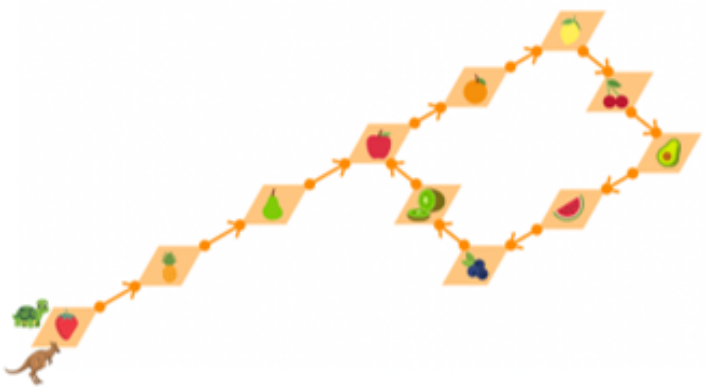
0 - 1 - 2 - 2 - 1
1 - 2 - 1 - 1 - 1
2 - 1 - 1 - 0 - 2
2 - 1 - 1 - 2 - 1
1 - 2 - 1 - 0 - 1
0 - 1 - 2 - 2 - 1

Buna göre yandaki görselde verilen ışıklı led panonun 5. satırını oluşturabilmek için hangi şifreleme yapılmalıdır?



A) 2 - 1 - 0 - 0 - 1
B) 1 - 2 - 2 - 0 - 0
C) 1 - 2 - 2 - 1 – 1
D) 2 - 1 - 1 - 0 - 0

11) Bir kanguru ve kaplumbağa aşağıdaki parkurda yarışmaktadır. İkisi de yarışa içinde çilek olan alandan başlayacak ve okları takip ederek bir sonraki alana ilerleyecektir. Aynı sürede kaplumbağa bir alan ilerlerken, kanguru iki alan ilerlemektedir. Örneğin, kaplumbağa ananas bulunan alana vardığında, kanguru armut bulunan alana varmaktadır.

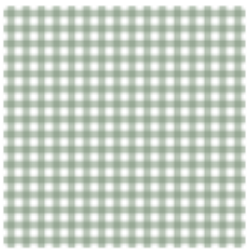


Buna göre kanguru ve kaplumbağanın başlangıç noktasından sonra ilk kez buluştuğu alanda hangi meyve bulunmaktadır?

A) Elma
B) Limon
C) Kiraz
D) Karpuz

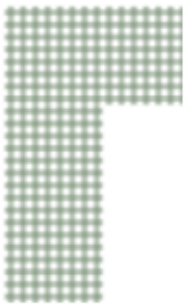
Figure C.7: CTST questions – page 7

12) Terzi Ayşe Hanım elindeki kare kumaşın tamamını parçalara ayırıp bu parçaları birleşme noktalarından dikerek yeni bir tasarım oluşturacaktır.



Terzi Ayşe Hanım, oluşturduğu tasarımda yalnızca bir yeri diktiğine göre, yaptığı tasarım aşağıdakilerden hangisi olabilir?

A)



B)

C)

D)

Figure C.8: CTST questions – page 8

13) Gülsüm elindeki 6 kibriti kullanarak aşağıdaki şekli oluşturmuştur. Yusuf bu şekildeki 1 kibriti hareket ettirerek yeni bir şekil oluşturacaktır.

Buna göre, aşağıdakilerden hangisi Yusuf'un elde edebileceği şekillerden birisi olamaz?

A)

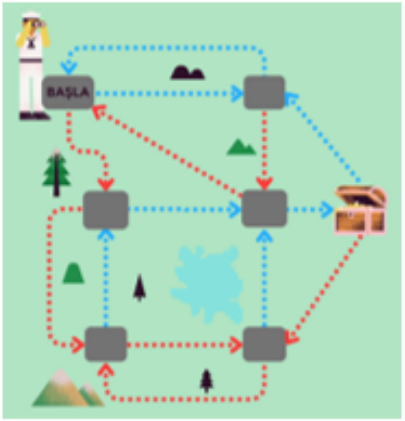
B)

C)

D)

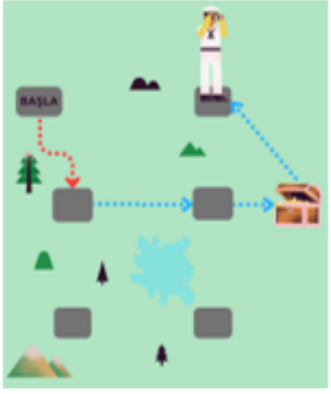
Figure C.9: CTST questions – page 9

14)



Denizci Ahmet, elindeki haritayı takip ederek hazineye ulaşmak istiyor. Denizci Ahmet, "Başla" yazan yerden başlayacak ve her seferinde bir yüzü kırmızı (K) bir yüzü mavi (M) olan bir madeni parayı havaya atıp tutarak, paranın üzerindeki renge göre ilerleyecektir. Ahmet parayı attığı durumda oluşan tüm basamakları ilerlemek zorundadır. Örneğin, parayı 4 kere attıysa, 3. adımda hazineye ulaşsa bile 4. adımı da ilerlemesi gerekmekte ve böylece hazineyi es geçmektedir.

Örneğin, Denizci Ahmet'in her adımda parayı atması sonucu oluşan sonuç KMMM ise buna göre Denizci Ahmet'in ilerlemesi sağdaki resimde gösterilmektedir.



Buna göre Denizci Ahmet, hangi sıra ile ilerlerse son adımında hazineye ulaşabilir?

A) KKMKMMK
B) MKMMMK
C) KKKMKKMM
D) KMKKMKKMK

Figure C.10: CTST questions – page 10

15) Arif'in bir kitaplığı vardır ve kitaplarını aşağıdaki kurallara göre düzenleyecektir:

1. Aynı türdeki kitaplar yan yana dizilecektir.
2. En fazla sayıda kitap içeren tür, kitaplıkta en başta yer alacaktır. En az sayıda kitap içeren tür ise en sonda yer alacaktır.
3. Aynı türe ait kitaplar, basım yıllarına göre en eskiden en yeniye doğru, baştan sona sıralanacaktır.

Aşağıda tabloda, kitapların isimleri, türleri ve basım yılları verilmiştir.

Kitap İsmi	Şeker Dağı'nın Gizemi	Gökyüzü Bilginleri	Gizemli Saat Kulesi	Tarih Kitabı Çetesi	Fındık Diyarı	Ayın Arkadaşları
Türü	Macera	Bilim	Macera	Tarih	Macera	Bilim
Basım Yılı	2019	2008	2010	2018	2005	2012

Buna göre Arif, kitapları baştan sonra doğru hangi sırayla dizilecektir?

A) Gökyüzü Bilginleri, Ayın Arkadaşları, Fındık Diyarı, Gizemli Saat Kulesi, Şeker Dağı'nın Gizemi, Tarih Kitabı Çetesi

B) Fındık Diyarı, Gizemli Saat Kulesi, Şeker Dağı'nın Gizemi, Ayın Arkadaşları, Gökyüzü Bilginleri, Tarih Kitabı Çetesi

C) Fındık Diyarı, Gizemli Saat Kulesi, Şeker Dağı'nın Gizemi, Gökyüzü Bilginleri, Ayın Arkadaşları, Tarih Kitabı Çetesi

D) Ayın Arkadaşları, Gökyüzü Bilginleri, Fındık Diyarı, Gizemli Saat Kulesi, Şeker Dağı'nın Gizemi, Tarih Kitabı Çetesi

Bir sakız makinesi, sırasıyla kırmızı, mavi, yeşil, sarı şeklinde sakız vererek bunu bir döngü şeklinde devam ettirmektedir. Sakız makinesi belirli bir noktadan sonra bozulmuş ve bu noktadan itibaren en başa dönmüştür.

Sakız makinesini tamir etmek için sakızın bozulduğu adımı bulması gereken tamirci, 25. adımda makineden sarı sakızın düştüğünü tespit etmiştir.

16) Buna göre, makine kaçıncı adımda bozulmuş olabilir?

A) 8 B) 10 C) 16 D) 19

17) Makine tamir edilmeden çalışmaya devam ettirilirse, 30. adımda makineden hangi renkte sakız düşer?

A) Kırmızı B) Mavi C) Yeşil D) Sarı

Figure C.11: CTST questions – page 11

18) Öğrenciler, bir eğitim dönemi boyunca farklı performans görevlerinden puanlar almıştır. Öğretmen, her öğrencinin hangi görevden kaç puan aldığını, toplam puanlarını ve her performans görevi için öğrencilerin aldığı toplam puanları bir tabloya kaydetmiştir. Ancak, tabloda bazı notlar silinmiştir.

Öğrenci	Görev 1	Görev 2	Görev 3	**Toplam**
Serkan	20	8		42
Hatice	9	★	13	
Meryem		11	14	40
İsmail	11			29
Toplam		43	45	

Yukarıda verilen tabloya göre, Hatice 2. performans görevinden kaç puan almıştır?

A) 10 B) 14 C) 15 D) 3

19) Zeynep, Mustafa ve Ali kırtasiyeden sevdikleri çıkartmaları alıyor.
Zeynep ve Mustafa kırtasiyeye birlikte gittiklerinde *araba, futbol, meyve, bitki, hayvan ve uzay* çıkartmaları almışlardır.
Mustafa ve Ali birlikte kırtasiyeye gittiklerinde *futbol, meyve, sebze, gülen yüz, hayvan ve kıyafet* çıkartmaları almışlardır.
Bu bilgilere göre, Zeynep, Mustafa ve Ali'nin en sevdiği çıkartmalar hangisi olabilir?

A) Zeynep: araba, futbol, bitki
Mustafa: sebze, gülen yüz, kıyafet
Ali: meyve, hayvan, uzay

B) Zeynep: futbol, meyve, hayvan
Mustafa: araba, bitki, uzay
Ali: sebze, gülen yüz, kıyafet

C) Zeynep: araba, bitki, uzay
Mustafa: futbol, meyve, hayvan
Ali: sebze, gülen yüz, kıyafet

D) Zeynep: araba, bitki, hayvan
Mustafa: futbol, meyve, uzay
Ali: sebze, gülen yüz, kıyafet

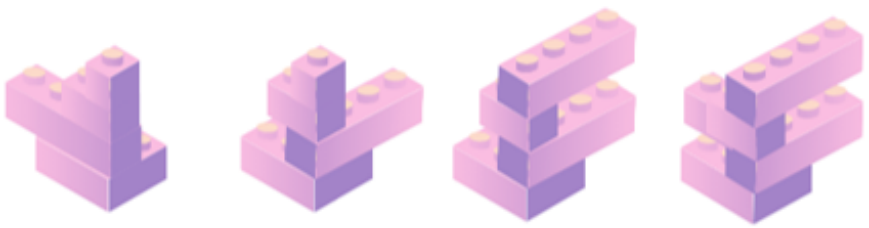
Figure C.12: CTST questions – page 12

20)

Yukarıdaki görselde verilen örüntüye göre soru işareti yerine aşağıdakilerden hangisi gelmelidir?

A) B) C) D)

21) Melike farklı büyüklükteki yapboz parçalarını farklı yönlerde takarak aşağıdaki dört şekli oluşturmuştur. Soldan sağa doğru oluşturduğu her şekil bir kelimeyi, her parça bir harfi sembolize eder. Kelimelerin ilk harfi en alttaki parçayı, en üstteki parça kelimenin sonuncu harfini temsil etmektedir.

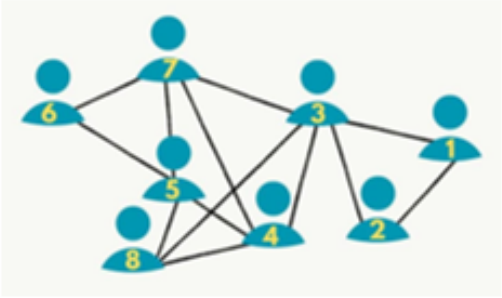


Melike'nin oluşturduğu kelimeler sırasıyla aşağıdakilerden hangisi gibi olabilir?

A) KALE-KİLE-KÖLE-KÖSE
B) KART-KURT-KURU-KUZU
C) SARI- SAPI-KAPI-KATI
D) SAKA-TAKA-TAKI-MAKİ

Figure C.13: CTST questions – page 13

Aşağıdaki harita Dilara'nın iş arkadaşı ağını göstermektedir. Haritada kişiler ve kişiler arasındaki bağlantıyı gösteren çizgiler bulunmaktadır. İki kişi arasında çizgi varsa bu kişiler iş arkadaşıdır. Haritada Dilara haricinde Elif, Zeynep, Mehmet ve Kemal de bulunmaktadır.



Yukarıdaki haritayla ilgili aşağıdaki bilgiler veriliyor:

- Elif, Zeynep ve Mehmet'in her birinin dört iş arkadaşı vardır.
- Zeynep ve Mehmet, Kemal ile iş arkadaşıdır.
- Kemal'in başka iş arkadaşı yoktur.

22) Buna göre yukarıdaki haritada Elif kaç numarayla gösterilmektedir?

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

23) Henüz Kemal ile iş arkadaşı olmayan Dilara, Mehmet aracılığıyla tanışacağına göre, Dilara haritada kaç numarayla gösteriliyor olabilir?

A) 1
B) 2
C) 5
D) 8

24) Okuldaki Çevre Kulübü bir geri dönüşüm projesi başlatmıştır. Her öğrenci getirdiği malzemeleri geri dönüştürebilmek için aşağıdaki kurallara uymalıdır:

- Her kutu en fazla 10 malzeme alabilir.
- Bir kutunun geri dönüşüm işlemi toplam 5 dakika sürmektedir.
- Geri dönüşüm işleminin aynı anda gerçekleştirilebildiği 3 masa vardır.
- Her masada aynı anda yalnızca bir öğrenci çalışabilir.
- Öğrenciler geliş saatlerine göre geri dönüşüm işlemine başlarlar.

Aşağıdaki tabloda, Ayşe, Mehmet, Zeynep ve Kadir'in dönüştürecekleri malzeme sayıları ve geliş saatleri gösterilmiştir.

Öğrenci	Malzeme Sayısı	Geliş Saati
Ayşe	8	12.05
Mehmet	14	12.00
Zeynep	6	12.00
Kadir	12	12.10

Buna göre aşağıdaki ifadelerden hangisi yanlıştır?

A) Mehmet geri dönüşüm işlemini 12.10'da bitirecektir.
B) Tüm geri dönüşüm işlemi 12.20'de biter.
C) Toplamda 6 kutu malzeme dönüştürülür.
D) Kadir geri dönüşüm işlemine başlayacağı esnada tüm masalar doludur.

Figure C.14: CTST questions – page 14

25) Yasemin odasına aşağıda verilen süslemeyi yapmak istiyor. Ancak süslemenin bir parçasını kaybetmiş.

Süslemeyi tamamlaması için soru işareti yerine aşağıdaki parçalardan hangisi gelmelidir?

A) B) C) D)

26) Her bir yüzü farklı bir renge boyanmış bir küpün üç farklı yönden görünümü aşağıda verilmiştir.

Buna göre aşağıdaki görünümlerden hangisi bu küpe ait <u>olamaz?</u>

A) B)

C) D)

Figure C.15: CTST questions – page 15