BARU Bartın University Journal of Faculty of Education
BUEFAD Volume 14, Issue 3
858-882
dergipark.org.tr/buefad
DOI: 10.14686/buefad.1670825

The Impact of Visual and Physical Programming-Based Science Activities on Middle School Students' 21st Century and Computational Thinking Skills

Habibe Güneş [a]*, ***Sevda Küçük*** [b]

a Dr. Research Assistant, Firat University, https://orcid.org/0000-0002-3479-2195 *habibekazezbote@gmail.com
b Professor, Ataturk University, https://orcid.org/0000-0002-2679-5177

Research Article
Received: 6.4.2025
Revised: 12.6.2025
Accepted: 20.6.2025

Abstract

In the age of artificial intelligence, one of the main goals of education today is to produce innovative and productive individuals with 21st century skills, who can think creatively, solve problems, think critically, and have strong collaboration and communication skills. This study examines the impact of STEM-based lesson plans that incorporate knowledge-based life problems (APoKS) and visual-physical programming activities on middle school students' 21st century and computational thinking (CT) skills. The study involved 15 middle school students enrolled in a summer course and three science teachers. A 30-hour intervention was delivered over two weeks, covering topics such as the solar system, force and motion, renewable energy, electrical circuits and sound. Data were collected using the Computational Thinking Skills Self-Efficacy Perception Scale and the 21st Century Skills Scale. After implementation, educators provided insight through semi-structured interviews and reflective diaries. Quantitative data were analyzed using the Wilcoxon Signed Rank Test, while qualitative data were examined using content analysis. Results indicate that the activities significantly improved students' 21st century thinking and CT self-efficacy. High impact improvements were observed in algorithm design, problem solving, data processing, programming and confidence. Educators confirmed these findings, noting the development of students' 21st-century and CT skills. Recommendations for future implementation and research are provided based on the findings.

Keywords: Programming, STEM, 21st century skills, computational thinking, self-efficacy, secondary school students

Görsel ve Fiziksel Programlamaya Dayalı Fen Etkinliklerinin Ortaokul Öğrencilerinin 21. Yüzyıl ve Bilgi İşlemsel Düşünme Becerileri Üzerindeki Etkisi

Öz

Yapay zekâ çağında günümüz eğitiminin temel hedeflerinden biri 21. yy becerilerine sahip yaratıcı düşünebilen, problem çözebilen, eleştirel düşünebilen, işbirliği ve iletişim becerileri kuvvetli, yenilikçi ve üretken bireyler yetiştirmektir. Bu çalışmada fen bilgisi eğiticilerinin hazırladıkları Bilgi Temelli Hayat Problemleri içeren FeTeMM'e dayalı ders planları ile görsel ve fiziksel programlama etkinliklerinin ortaokul öğrencilerinin 21.yy ve bilgi işlemsel düşünme (BİD) becerilerine etkisinin incelenmesi amaçlanmaktadır. Çalışma grubunu yaz kursuna katılan 15 ortaokul öğrencisi ve 3 fen bilgisi eğiticisi oluşturmaktadır. Uygulama iki haftada 30 saatlik müdahaleyle gerçekleştirilmiştir. Uygulamada ortaokul fen bilgisi dersi konularından güneş sistemi ve ötesi, kuvvet ve hareket, kuvvet ve enerji, yenilenebilir enerji, elektrik devreleri ile ses ve özellikleri konularında etkinlikler gerçekleştirilmiştir. Veri toplama aracı olarak Bilgi İşlemsel Düşünme Becerisi Öz-yeterlik Algısı ölçeği ile Ortaokul Öğrencilerine Yönelik 21. Yüzyıl Becerileri Ölçeği kullanılmıştır. Uygulama sonrasında eğiticilerden yarı yapılandırılmış görüşme ve yansıtıcı günlükler yoluyla ortaokul öğrencilerinin 21. yy ve BİD becerilerine ilişkin görüşleri alınmıştır. Nicel veriler Wilcoxon İşaretli Sıralar testiyle analiz edilmiş, nitel veriler ise içerik analizi yöntemiyle incelenmiştir. Bulgulara göre etkinliklerin ortaokul öğrencilerinin 21.yy düşünme becerileri ve bilgi işlemsel düşünme (BİD) öz yeterlik algısı üzerinde anlamlı bir fark oluşturduğu görülmüştür. Öğrencilerin 21. yy becerilerinde yüksek etki düzeyinde bir değişim olduğu, BİD öz yeterlik algılarında ise algoritma tasarlama, problem çözme, veri işleme, temel programlama ve özgüven boyutlarının hepsinde yüksek etki düzeyinde gelişme olduğu göze çarpmaktadır. Eğiticiler de nicel bulguları destekleyen şekilde öğrencilerinin 21. yy becerileri ve BİD becerilerinin uygulama sonrası geliştirdiğini düşünmektedirler. Araştırmadan elde edilen bulgular doğrultusunda uygulama sürecine ve gelecek araştırmalara yönelik öneriler sunulmuştur.

Anahtar Kelimeler: Programlama, FeTeMM, 21. yy becerileri, bilgi-işlemsel düşünme, öz yeterlik, ortaokul öğrencileri

To cite this article in APA Style:

Güneş, H., & Küçük, S. (2025). The impact of visual and physical programming-based science activities on middle school students' 21st century and computational thinking skills. *Bartın University Journal of Faculty of Education, 14*(3), 858-882. https://doi.org/10.14686/buefad.1670825

*This article is derived from the first author's PhD dissertation completed in 2023 under the supervision of the second author.

*This study was presented as an abstract with the title "The Effect of Programming-Based Science Activities on Secondary School Students' 21st Century and Computational Thinking Skills" at EJER conference on June 8-11, 2023.

INTRODUCTION

In the age of artificial intelligence, the welfare level of countries is measured depending on their strength in the cyber environment, their economic stability and their ability to produce their own technologies in terms of scientific developments (Davenport, 2018). According to the concept that examines the impact of technology on the labour force based on Schumpeter's concept of 'creative destruction', innovation will lead to the decline of old technologies and sectors, causing job losses and sectoral transformations (Aşkun, 2024; Yavuz-Aksakal & Ülgen, 2021). Within the scope of this transformation, in the 6th wave covering the period between 2020-2060, individuals who can write their own codes and work in harmony with 21st century skills stand out as digital citizens that countries want to prepare for the future, especially in the artificial intelligence era (Eteng et al., 2022; Yılmaz & Yılmaz, 2023). Accordingly, education systems have to be restructured to educate individuals who are entrepreneurial, innovative and can use technology effectively (Çiftci et al., 2021; Kazakoff et al., 2013; Perkovic & Settle, 2010). Because most 21st century students are still being educated in 19th century school organisations with 20th century pedagogical approaches (Amadi, 2022; Schleicher, 2018). This situation reveals that education systems have difficulty in keeping up with the requirements of the digital age and teaching methods need to be updated (Yalap & Gazioğlu, 2023). In order for digital transformation to be effective in education, it is of great importance to restructure education policies and teaching strategies in an innovative way with a planning based on a system approach (Bozkurt et al., 2021). When the reports that shape education policies are analyzed, it is seen that the competencies expected from learners have undergone a significant transformation in the last 20 years. While in 1998, the basic skill that should be taught to students was defined as 'learning to use technology', as of 2007, this approach has been transformed into 'using technology to learn'. Because the 21st century is quite different from the 20th century in terms of the skills that people need for work, citizenship and self-realisation (Dede, 2010). By 2016, it is emphasised that students should exhibit 'transformative learning with technology' skills (ISTE, 2023). The implementation of digital transformation in education in a planned manner and on the basis of a system approach is critical in terms of reshaping learning-teaching processes in an efficient and effective way (Bozkurt et al., 2021). Because the system approach aims to ensure that the elements that interact with each other work together and that the problem experienced in any part is solved without affecting the whole system. This principle is important for the successful implementation of digital transformation in education, because the difficulties encountered in the digital transformation process need to be solved quickly and made more efficient without damaging the overall functioning of the system. One of the most important approaches supporting digital transformation in education has been STEM (Science, Technology, Engineering and Mathematics) education. Since 2001, this interdisciplinary approach, which has become widespread, aims to provide students with the ability to solve real-world problems (Bybee, 2013; Cheng et al, 2021). Thanks to STEM education, students have the opportunity to develop critical thinking, problem solving, creativity and collaboration skills by using their academic knowledge practically (Çorlu et al., 2014). This approach overlaps with John Dewey's view that disciplines should not be separated from each other and that students should receive an experience-based education in which they interact with the ever-changing world (Johnson & Reed, 2008; Sublette, 2013). In fact, although this approach has become widespread in the 21st century, although it is not named with such a framework, when the fields of study and inventions developed by Turkish-Islamic scholars are examined, it is seen that they specialised in more than one field in many branches such as mathematics, astronomy, medicine and physics. Considering the solutions found by Al-Jazari, the pioneer of robotics, to the real problems of the day by combining engineering, mathematics, geometry and art, it is possible to say that the first examples of the STEM approach were also used in the 12th century (Polatgil, 2020).

Today, the integrated teaching framework is accepted as a theoretical roadmap for STEM education practitioners, teacher educators and researchers in developing 21st century skills and competencies (Çorlu & Çallı, 2017; Hoeg & Bencze, 2017). Within the framework of Integrated STEM (I-STEM), knowledge-based life problems (APoKS) are placed at the centre. These problems have a structure that allows examining the interaction of multiple variables and encourages students to develop different solutions within certain limitations rather than directing them to a single correct solution (Başaran, 2018; Çorlu & Çallı, 2017). Therefore, in order to overcome such problems, individuals need to have 21st century skills. B-STEM education allows students to use their mathematics, science literacy skills and problem-solving skills and to develop their technological literacy by focusing on open-ended exploration and real-world problems in engineering design processes (Falloon, 2019; Honey et al., 2014). As a result, today's education systems should not only teach the use of technology, but also aim to raise students as individuals who can use computer science effectively, enrich learning processes with technology, and learn by critically filtering information (Bers, 2019; Mezirow, 1996). Innovative educational approaches such as B-STEM provide a radical transformation in education systems in line with this goal, while eliminating inequalities in computer education for all (Weintrop et al., 2014). It is also an effective approach in developing positive attitudes towards science (Koca, 2018).

21st Century Skills and Sub-Dimensions

In many sources for 21st century skills, the concepts of 4C (Communication, collaboration, critical thinking, creativity) as communication, collaboration, critical thinking and creativity are mentioned (P21, 2019). However, according to Mazzola-Randles (2020), connectedness, as a fifth dimension, is now considered among these skills. The connectivity dimension includes qualities that learners will need in the 4th industrial revolution, such as digital well-being, commitment to identity learning, developing digital content, and building and maintaining communities (Mazzola-Randles, 2020). This concept implies that the learner connects to networks in line with their own learning needs or creates their own learning networks, that learning is not uniform and linear, and that non-linear asymmetric learning approaches are used in digital environments and online networks today. Creativity, on the other hand, is defined as thinking outside traditional ways of thinking, challenging one's own skills and abilities, nurturing a sense of curiosity, using imagination and being productive (Voogt & Robin, 2012).

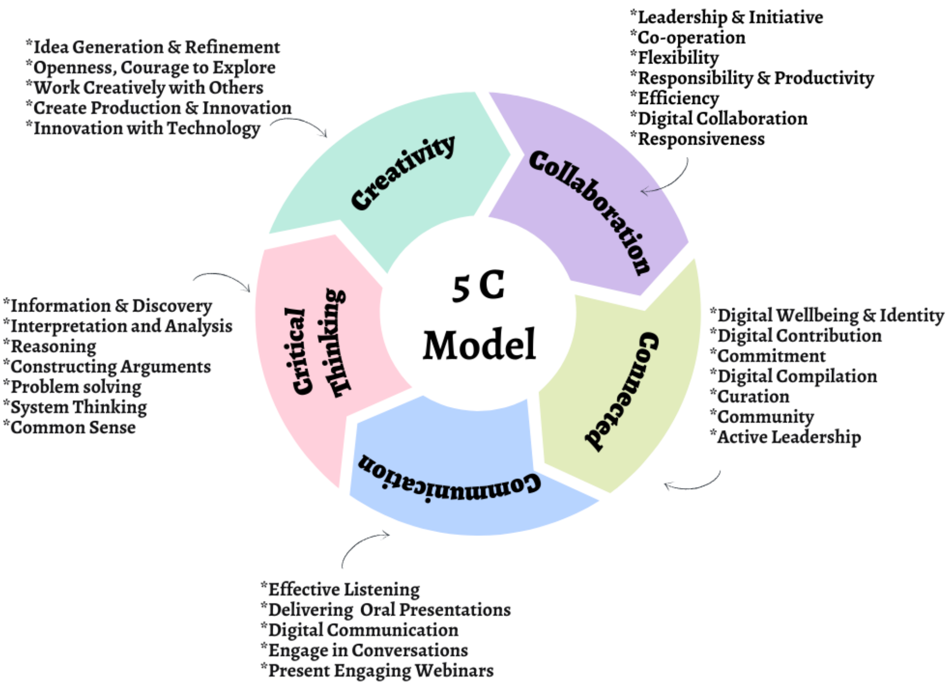

Figure 1. *21 st Century Skills and Sub-Dimensions*

Critical thinking is a skill that increases the self-development and autonomy of the individual, including self-management, self-organisation, self-regulation, self-direction, self-evaluation, independent thinking, autonomous action and management skills (Facione, 1990; Mete, 2021). Critical thinking is a competence that promotes analytical thinking, problem-solving abilities and high-level thinking skills, enabling the individual to defend his/her rights and use his/her emotional intelligence (Mazzola-Randle, 2020). It is important for the individual to be consistent in his/her thoughts, to distinguish between reliable and unreliable information while searching for the truth, to make judgements with sufficient evidence, and to perceive the relationships between the data obtained as a result of research and observations (Bülegin & İlkörücü, 2023). Although the importance of the family among environmental factors in the development of critical thinking is important, it is known that educational interventions especially at a young age, such as critical-based science activities, are effective in developing students' cognitive and affective skills (Bülegin & İlkörücü, 2023; Mete, 2021). In addition, it has been observed that the development of this skill has a strong predictive effect on students' academic achievement, and academic achievement and critical thinking are highly effective on mathematics achievement, which is a dimension of STEM, as the grade level increases (Alsancak & Aybek, 2023; Er, 2024). STEM-supported educational environments are a concept that includes teamwork, open-mindedness, conflict management, self-motivation, entrepreneurship, and leadership through interaction, especially in diverse and heterogeneous environments by supporting the dimension of cooperation and communication (Herdem & Ünal, 2018; Karakaya & Avgın, 2016).

Computational Thinking Skills and Sub-Dimensions

Computational thinking (CT), a 21st century skill, is a competence that provides positive contributions to interdisciplinary learning processes aiming to enable individuals to solve problems using technology (Gretter & Yadav, 2016; Güllü Eğin & Sözer, 2024; Tosik-Gün & Güyer, 2019). According to Wing (2006), computational thinking skill is a way of thinking that people of all ages should acquire as a skill that includes problem solving using the basic concepts of computer science, designing systems and understanding human behaviour using computer science concepts. Computational thinking skill, which is aimed to be developed at an early age by being integrated into the preschool programmes of countries, is now considered as one of the basic life skills such as arithmetic, reading and writing (Macrides et al., 2022; Xu et al., 2022; Uğraş et al., 2025). This skill, which is based on computer science and coding, contributes to the development of problem solving, analytical thinking, creative thinking and algorithmic thinking skills needed in mathematics and science (Küçükaydın et al., 2024; Yıldız-Durak et al, 2019).

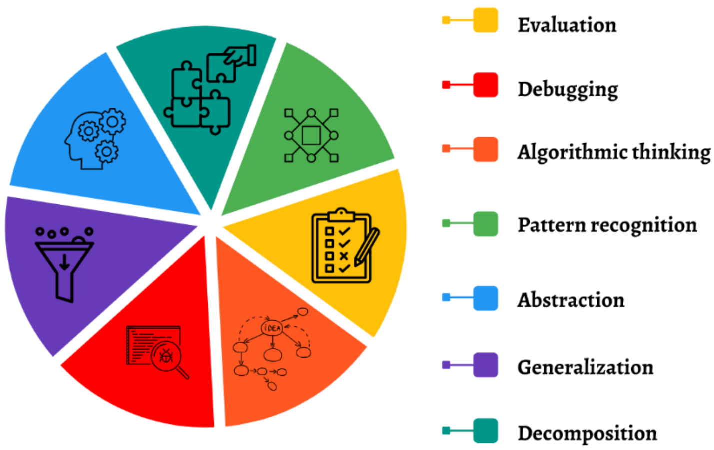

Figure 2. *Computational Thinking (CT) Skills and Sub-Dimensions*

In the literature, it is emphasized that applications that integrate coding and computational thinking are needed to raise individuals who can think computationally, solve problems and are open to innovations (Demir & Seferoğlu, 2017, Korkmaz et al., 2017). One way to provide individuals with these skills is programming and computer science teaching (Cross et al., 2016; Kert et al., 2020). In particular, teaching computer science at an early age is considered very important for children's cognitive skills, STEM subject learning, 21st century skills, and future career directions (Cantlon et al., 2024; Li et al., 2020). Today, artificial intelligence (AI)-supported STEM education, STEM+AI, is also used to improve learners' computational thinking skills and questioning ability (Li et al., 2023). Algorithmic thinking is a systematic problem-solving method consisting of well-defined and sequential steps that can be performed in a certain period of time (Kanaki & Kalogiannakis, 2022). Algorithmic thinking is the ability to create and apply algorithms to define and solve problems (Barr & Stephenson, 2011). This process involves dividing a problem into sub-steps and developing algorithms for each step. According to Wing (2006), algorithmic thinking is a concept that supports problem solving not only in computer science but in all fields, is closely related to CT and is often used interchangeably. However, CT is a broader concept that includes algorithmic thinking as well as other skills such as data representation, modeling, simulation and debugging. Abstraction refers to identifying the basic concepts and data needed to solve complex problems, extracting similarities and ignoring unnecessary elements (Kert et al., 2020). Abstraction includes stages such as data collection, pattern recognition, and modeling (Shute et al., 2017) and enables the elimination of unnecessary details in achieving the goal. Decomposition is the process of breaking a problem into smaller and manageable sub-problems (Barr & Stephenson, 2011). This method enables solving complex problems, detecting bugs and developing reusable, modular code (Kelleher & Pausch, 2005). This approach plays a critical role in algorithm design, helping to generate effective solutions to large and complex problems. Evaluation is the process of testing the accuracy, efficiency and applicability of the solutions developed. It determines whether an algorithm or program produces the desired results and whether it is the best solution. Error detection and debugging play an important role in this process. Debugging is to ensure that the program works correctly by correcting the detected errors. These two processes contribute to effective problem-solving by increasing the accuracy and reliability of programs and algorithms. Pattern recognition enables the algorithms developed to be configured on computers and applied to different problems (Cansu & Cansu, 2019). While increasing efficiency, it reduces repetitive tasks and reduces the probability of error. Generalization is the process of adapting a solution to similar problems by reusing its components. It involves creating general solutions that can be applied to a wider set of problems. This skill is important not only in programming but also in general problem solving processes. Various platforms are often used in computer science education to support the development of computational thinking skills. Various platforms (e.g., Alice, Blockly, Code Org, Scratch, Mblock, and KoduLab) are used for visual programming tools, and physical programming tools such as Lego, Robotis, and VEX are used for physical programming with robotics education kits containing microcontrollers, various sensors, gears, and motors (Gunes & Kucuk, 2022; Selcuk et al., 2024). Unlike text-based programming tools with complex code structures, block-based platforms facilitate learning algorithms and programming for young learners due to their drag-and-drop nature (Bers, 2019; Kert et al., 2020). The use of robotics in education is an approach that has become increasingly common in recent years (Hangün et al., 2022; Kazez & Genç, 2016). When the educational outcomes were examined in more detail, it was seen that there was a positive change in the development of students' academic achievement (Yılmaz, 2019), spatial skills (Coxon, 2012; Julià, & Antolí, 2016; Koca, 2020; Şişman et al., 2021). The use of robotics in education has positively increased the motivation of learners in the learning process (Şişman & Küçük, 2019) and enabled students to form positive perceptions about science and technology courses, robotics and programming and academic self-concept (Karaahmetoğlu, 2019; Kardeş, 2020; Özel, 2019). Unlike other computer technologies, robotic technologies are not an activity in which children sit alone in front of a technological tool such as a computer, but a technology that develops their computer thinking, improves their motor skills and hand-eye coordination through socialization (Yıldız-Durak et al., 2019). In addition, teaching programming at an early age makes it easier for learners to concretize abstract concepts in mathematics such as number, size and shape, and improves their language skills and visual memory (Nugent et al., 2008). In addition to these, considering that the career preferences of female students in STEM fields are low in the world, it is known that educational robotics contributes to the motivation of female students that they can be successful in computer and engineering sciences and make them pursue a career in these fields, and it has positive contributions to increase girls' skills in information technologies (Seraj et al., 2019; Polat et al., 2021). In science education, ICT skills, including problem solving and logical inquiry concepts, are important for the effectiveness of science teaching (Arslanhan & Artun, 2021; Wang et al., 2022). In addition, for a more effective education, it is of great importance that the lesson plans prepared by teachers are designed to develop students' 21st century skills and computational thinking competencies and that these lessons are implemented in a way to increase learners' technological competencies (Gezer & Durdu, 2025). Educational interventions that include STEM-based APoKS not only develop students' 21st century skills but also enable them to solve real-life problems by using science, technology, engineering, and mathematics in an integrated manner through social participation (Cheng et al., 2021).

However, although there are various computer science initiatives such as Computer Science for All (CT for All) by the International Society for Educational Technology and the Association of Computer Science Teachers in the UK, and the National Computer Science Study Programs by the Computer Science Teachers Association, there is still much unknown about how best to support the integration of computer science into classroom instruction (Ketelhut et al., 2020; Kong et al., 2023). Moreover, the using information technology (IT) in science classes is an effective alternative to improve students' IT skills (Ogegbo & Ramnarain, 2022; Zakwandi & Istiyono, 2023). The problem-oriented nature of science practices enables the inclusion of ICT in the problem-solving process and the implementation of this process in a structured way (Zakwandi & Istiyono, 2023). In particular, the science learning process that emphasizes discovery through experimental activities can be associated with dimensions of ICT such as problem identification, decomposition, data collection, algorithm creation and generalization (Sarı & Karaşahin, 2020). In the literature, there are various suggestions and applications for the integration of ICT into science classrooms, but these are usually computerized coding applications and lack traditional science education (Ogegbo & Ramnarain, 2022). However, especially in science education, it is seen that 22-hour Arduino-supported robotic coding activities carried out in the 5th grade at the middle school level contribute to students' positive attitudes towards the course and increase their attitudes towards the use of technology in lessons (Güven & Sülün, 2023). In addition, it was found that students' academic achievement increased with 16 hours of intervention in the Force and Motion unit in 6th grade middle school courses (Gümüş & Eroğlu, 2024; Uçar & Sezek, 2024). Since more studies have been conducted on meaningful learning and attitude in science education (Gümüş & Eroğlu; 2024; Soypak & Eskici, 2023), it is thought that examining the effect of a learning process supported by visual and physical programming activities in middle school science education on learners' 21st century skills and CT self-efficacy will contribute to the literature.

The Significance of the Study

It is stated that educational tools and environments that support 21st century skills are insufficient in the education programs implemented in Turkey, and learning objectives are not clearly defined (Kurudayıoğlu & Soysal, 2019). Especially in PISA reports, when Turkey's achievement average is analyzed by years, it is seen that there is a significant decline in the field of science, so educational interventions in this field are important (Aydın & Çilek, 2024). Bozkurt and Çakır (2016) found that students' 21st century skills decreased as their grade levels increased and did not develop sufficiently during the teaching process. It is also emphasized that teachers lack knowledge on how to develop and measure these skills, which prevents students from acquiring these skills (Yalçın, 2019). In particular, it was stated that the activities supporting creativity are limited and the current curriculum is insufficient to develop students' creative thinking skills (Keleşoğlu, 2017; Yurdakal, 2018). In this context, curricula should be prepared to support active, collaborative, project-based and student-centered approaches. It has been observed that even short-term robotic interventions involving programming, especially in summer courses, improve students' computational thinking skills and increase their self-efficacy in rural areas where students experience educational inequality (Shang et al., 2023). Similarly, in an experimental study, Pellas (2024) reported that summer course activities involving robotics and programming with concrete programming tools also improved abstraction, problem decomposition, and visuospatial reasoning skills in preschool children. In addition, it is noteworthy that programming activities with visual and physical programming tools are mostly used in scientific process skills and problem solving skills of learners in science courses, but there is a limited number of studies for computational thinking and 21st century skills (Authors, 2022). For this reason, in this study, science lesson plans including visual and physical programming activities for middle school students were implemented in a summer course and their effects on students' 21st century skills and computational thinking skills were examined. Thus, it is aimed to contribute to the elimination of the current uncertainty by aiming to understand the effect of computational thinking on technology integration more clearly. It will be investigated how to integrate computational thinking skills not only at the level of programming instruction and individual self-efficacy, but also how to integrate them more effectively into classroom practices. Furthermore, concrete data will be obtained on how STEM-based educational approaches contribute to developing students' 21st century skills and supporting their ability to solve real-world problems. In this context, it is planned to provide a new perspective on how technology integration can be adapted to teaching processes and how students can be more effectively involved in these processes. The study is expected to provide strategies and methods to support the effective integration of computer science teaching in the classroom and to eliminate current uncertainties.

In this study, science lesson plans including visual and physical programming activities for middle school students were implemented and their effects on students' 21st century skills and computational thinking skills were examined. In this context, it was aimed to understand the effect of computational thinking on technology integration more clearly and to contribute to the elimination of existing uncertainties. The study examined how computational thinking skills, which are only addressed at the level of programming instruction and individual self-efficacy, can be integrated more effectively in classroom practices. It also provided concrete data on how STEM-based educational approaches contribute to developing students' 21st century skills and supporting their ability to solve real-world problems. In this context, the purpose of this study is to examine the impact of educator-developed visual and physical programming activities on middle school students' 21st century skills and computational thinking skills. The research seeks answers to the following questions:

1) Is there a significant difference between the pre-test and post-test scores of students regarding 21st century skills?

2) Is there a significant difference between the pre-test and post-test scores of students' self-efficacy perceptions of computational thinking skills?

3) According to educators;

a) What is the impact of programming-based science activities on students' 21st century skills?

b) What is the impact of programming-based science activities on students' self-efficacy perceptions of computational thinking skills?

c) What is the impact of programming-based science activities on students' challenges encountered in implementation and suggestions for implementation?

This study aims to provide a new perspective on how technology integration can be adapted to teaching processes by revealing the effects of educational activities on students' skill development.

METHOD

Research Design

In the study, explanatory design, one of the mixed methods researches, was used to examine the effect of science activities based on visual and physical programming on the 21st century and computational thinking (CT) skills of middle school students. The aim of mixed methods research in which qualitative and quantitative research methods are used together is to reduce the limitations that may arise from the use of only one of the research methods, to obtain more comprehensive data and to strengthen the findings (Creswell, 2013; Fırat et. al, 2014).

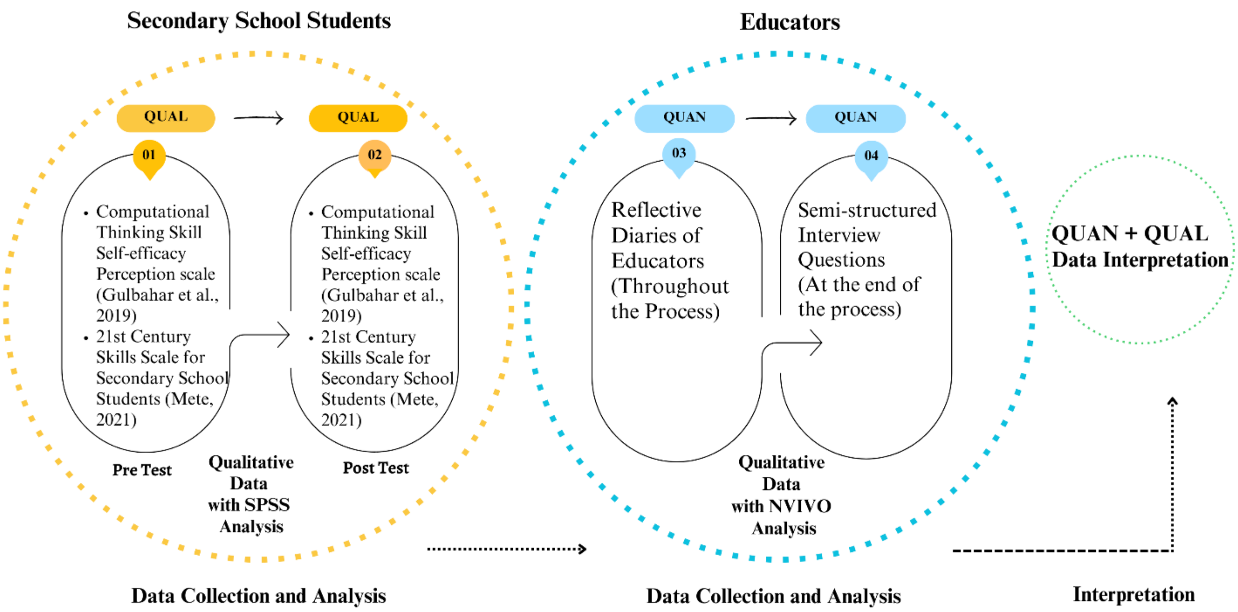

Figure 3. *Mixed Methods Research Explanatory Design*

Two stages of the explanatory design were followed in the study. As seen in Figure 3, in the first stage, the quantitative data obtained from the study group by applying the pre-test and post-test were analyzed in order to determine the 21st century and computational thinking skills of the students. Then, qualitative data were used to explain the quantitative results obtained in the first stage. The results were obtained by interpreting both quantitative and qualitative findings together.

Study Group

The study group, which was selected by purposive sampling method, consisted of 15 secondary school students and three educators (*E1, E2, E3*). The improvement in students' 21st-century and computational thinking skills was elaborated through qualitative data gathered from the insights of the educators. The educators, to whom the first author provided training on visual and physical programming as a coordinator instructor, are studying in the department of science teaching and are experts in visual and physical programming and 3D design. In this study, the development of students' skills was reported from the perspective of pre-service teachers based on their direct classroom experiences and observations throughout the intervention process. The secondary school students in the study group, 60% (n=9) were male and 40% (n=6) were female. Sixty per cent of the students (n=9) completed the 5th grade and 40% (n=6) completed the 6th grade. It was determined that only 20% (n=3) had experience in visual programming and the remaining 80% (n=12) had not participated in any study in this field. It was observed that none of the students had experience in 3D modelling. While 33.3% (n=5) of the students had experience in physical programming, 66.7% (n=10) did not participate in any study in this field. Within the scope of Arduino activities, 33.3% (n=5) of the students performed ‘Led Experiment’, but 66.7% (n=10) did not have any Arduino experience. The rate of students with course experience was found to be 26.7% (n=4), while 73.3% (n=11) had not attended any course before. It is seen that the students mostly had no previous experience in visual and physical programming, and the students who stated that they knew about it encountered basic level activities. The students voluntarily decided which group they wanted to be in and determined a group name. The Civcivler group mainly consisted of male and female students who had completed the 5th grade and were studying at a public school. The Magnafen group is a group of four students, all female students attending a private school. The Hababam group is a group of six male students who have completed the 5th or 6th grade and attend public and private schools.

Data Collection Tools & Process

The implementation was planned as a summer school and was carried out in a 10-day intervention (30 hours) of three hours each lasting two weeks. In the implementation, activities based on visual and physical programming were carried out on the subjects of solar system and beyond, force and motion, force and energy, renewable energy, electrical circuits, and sound and its properties from the 5th, 6th and 7th grade science course subjects. The Computational Thinking Skill Self-Efficacy Perception Scale, which was developed by Gülbahar, Kert and Kalelioğlu (2019), was used in the study. The scale demonstrated strong validity, as indicated by a KMO value of .966 and a significant Bartlett's test ($p <.05$), as well as high reliability, with item-total correlations ranging from .386 to .632 and Cronbach's alpha coefficients between .762 and .930. The study also utilized the 21st Century Skills Scale for Secondary School Students developed by Mete (2021). This scale demonstrated strong validity with a KMO value of .954, a significant Bartlett’s test ($p < .05$), and a confirmed factor structure through exploratory factor analysis, alongside high reliability evidenced by a Cronbach’s alpha of .81 and a test-retest correlation of .72. In addition, after the implementation, semi-structured interviews and reflective diaries were conducted with the educators who carried out the science activities based on visual and physical programming to obtain their opinions on the 21st century and ICT skills of middle school students. Due to the small sample size, quantitative data were analysed with Wilcoxon Signed Ranks test, one of the non-parametric tests, and qualitative data were analysed with content analysis method.

Validity and Reliability

This study adopted a mixed methods approach, collecting data from multiple sources to enhance validity and reliability (Topu et al., 2014). In the quantitative phase, the validity of the data collection instruments was evaluated in relation to the literature, and expert opinions were consulted during the development of new instruments. The rationale for the selection of the methods used was explained in detail, and instructional materials were prepared in alignment with the learning outcomes of the middle school science curriculum set by the Ministry of National Education. The consistency of the data was checked, and the reliability of the quantitative instruments was ensured. In the qualitative phase, the characteristics of the study group and the process of its selection were described, and both the implementation process and the researcher’s role were elaborated. During data collection, participants’ voluntary consent was obtained, and triangulation was employed. Inter-coder agreement was calculated during the analysis of qualitative data, and necessary measures were taken to ensure the validity and reliability of the data collection tools.

Procedure

In the study, each educator taught two lessons based on the lesson plans they prepared for 10 days and the flow of the process is given in Table 2. Each lesson lasted for three hours. On the last day of the first week, each of the educators guided a group while designing a Scratch project with the students. In the second week, on the last day of the week, the same group completed the Arduino experiment by guiding the students.

Table 1. Summer Course Activity Plan

	Subject	Coding Tool
Week I	• First Lesson Introductions, • Ice-breaker events, • Understanding students' prior knowledge • Implementation of pre-tests • Hour of Code activity • Introduction to Scratch	Hour of Code Scratch 3.0
	Solar System and Beyond	Scratch
	Force and Motion	Mbot ve Scratch

Week II	Force and Energy	Scratch
	First project: Science Projects with Scratch	Scratch
	Natural and artificial resources, Renewable Energy	Scratch
	Electric circuits	Arduino
	Sound and Features, 3D printer and features	Arduino
	Identification of project topics and planning	
	Completion of the selected Arduino experiment with groups, presentation and reporting to the class Implementation of post-tests	

The syllabus of the summer course was implemented with the lesson plans developed by the educators after the gains and sub-gains were determined in accordance with the MEB education programme. The educators followed the students during the process, observed the students who were suitable for working together and took notes. Since the students had no previous experience in robotics and programming, the first week of the activities was carried out with only visual programming and the second week with physical programming activities.

Table 2. Content of the Implementation Steps

Subject	Content
Hour of Code Scratch	• Students were introduced to visual programming via Hour of Code and earned multiple certificates. Basic Scratch concepts and structures were taught.
Solar System and Beyond	• Students created independent projects without guidance. They coded rotating planets and used text-to-speech features. Learning through exploration was encouraged.
Force and Motion	• Introduction to Arduino and Mbot components. Bluetooth pairing and testing on inclined planes. Students observed force requirements on different slopes and simulated them with Scratch.
Force and Energy	• Related to previous lessons using planetary force connections. Concepts like gravity and potential energy modeled in Scratch. Students created a scoring game and applied peer learning.
First Project	• Educators guided the groups during project development. • Students created materials in Scratch to address problems they identified. • Completed projects were shared and reviewed with classmates. • Civcivler and Hababam Groups: Modeled friction on different surfaces. • Magnafen Group: Explored the question “If Earth disappeared, which planet would be suitable?” and developed the “Life on Planets” project.
Renewable Energy	• Concepts introduced through instructor-prepared Scratch project. Students learned through gameplay and code analysis.
Electric circuits	• Introduction to circuits using Arduino and Tinkercad. Students tested different power sources and learned about resistance through experiments. Designed LED circuits (single, triple, ten-light). Experienced digital collaboration via virtual classrooms.
Sound and Features	• Explained sound energy and developed sensor-based fan systems. Discussed real-life applications (e.g., air quality monitoring). Designed whistles using 3D modeling and printing.
Identification of project topics and planning	• Students prepared including APoKS based projects on topics they selected themselves. Solutions were developed using materials from the Arduino kit. • Civcivler Group: Designed a motion-detecting alarm system (PIR sensor, light, and buzzer) to ensure home security. • Magnafen Group: Developed a natural gas alarm system (MQ-7 sensor, buzzer, and red LED) providing audio alerts for visually impaired individuals and visual alerts for those with hearing impairments. • Hababam Group: Created a parking sensor project (ultrasonic sensor and buzzer) to measure parking distances and reduce accidents.
Second Project	• Problems in the projects were solved collaboratively by students, enhancing their collaboration skills. • Coding knowledge was reinforced and incorrect codes were corrected to complete the projects. • Creativity was demonstrated, and various solutions were developed for the selected projects. • Projects were completed independently without guidance, showing self-directed learning skills. • Communication and presentation skills improved as students presented their projects in class. • Projects were evaluated by instructors, feedback was provided, and students’ progress throughout the process was observed.

As shown in Table 2, students created two group projects each in both visual and physical programming, guided by the educators throughout the process. The educators evaluated the projects prepared by the students within the framework of 21st-century skills and computational thinking skills. The projects titled Civcivler, Hababam, and Magnafen were analyzed in terms of critical competencies such as problem-solving, algorithmic thinking, creativity, collaboration, and independent work. During the evaluation process, feedback was provided based on the students' technical accuracy, innovative approaches, and engagement in the project development phases. In this way, the development of both cognitive and collaborative skills of the students was supported.

Figure 4. *Examples of Implementation*

Figure 4 presents examples showing that students engaged with topics such as force and motion, electrical circuits, and programming through experiential activities. Students explored the components of the Arduino-based Mbot, controlled it via Bluetooth, and investigated the relationship between force and incline through inclined plane experiments. Each student later raced their robot on behalf of their group. In the second image, a force variable was defined using Scratch to simulate Newton's law of gravity, and learning was supported through gamification. In the topic of electrical circuits, after introducing circuit components using Arduino and Tinkercad, the concept of resistance was demonstrated. By designing LED circuits, students experienced a sense of digital community within a virtual classroom environment.

Quantitative Data Analysis

Descriptive statistical tests were used to analyze the quantitative data in the study. Since the data did not show a normal distribution, the Wilcoxon test, one of the non-parametric tests, was chosen to understand the pre- and post-intervention situations. Descriptive statistical tests were used to analyze the quantitative data in the study. Before conducting the analyses, the normality of the data was assessed by examining skewness and kurtosis values. Additionally, since the number of observations was below 30, the Shapiro-Wilk test was conducted to further evaluate whether the data were normally distributed (Pallant, 2007). The results showed that while most of the pre- and post-test scores had p-values greater than .05, indicating normal distribution (e.g., 21st-century skills pre-test: $p = .630$; computational thinking pre-test: $p = .309$; post-test: $p = .093$), the 21st-century skills post-test score significantly deviated from normality ($p = .001 < .05$). Therefore, due to the violation of the normality assumption in this variable and to maintain consistency in statistical comparisons, the Wilcoxon signed-rank test—a non-parametric alternative—was used to analyze pre- and post-intervention differences. Fraenkel and Wallen (2009) defined effect size as a measure of the magnitude of the difference between the means of two groups. Cohen (1988) and Pallant (2007) stated that an r value between 0.1 and 0.3 indicates a small effect size, between 0.3 and 0.5 indicates a medium effect size, and 0.5 or above represents a large effect size.

Qualitative Data Analysis

Qualitative data were analyzed using the content analysis method, and NVIVO 11 was utilized to systematically analyze and organize the data. Themes, codes, and sub-codes were derived through an inductive approach by the researcher. To ensure reliability, the themes, codes, and sub-codes were reviewed in collaboration with an expert. Inter-coder reliability was calculated using the formula developed by Miles and Huberman (1994, p. 64), in which matching codes are categorized as "Agreement" and differing ones as "Disagreement." The reliability formula is: Agreement Percentage = Agreement / (Agreement + Disagreement) * 100. During the analysis of the qualitative data from the summer course, it was observed that the researcher and the expert reached consensus in cases of discrepancy, resulting in a high inter-coder agreement percentage of 96%.

Research Ethics

In this study, all ethical procedures have been followed. All participants have been informed about the purpose, process, and ethical rights of the research. All information collected was anonymised, confidential and only available to the researcher and her supervisor. Pseudonyms were used throughout the studies to replace educators' and students' real names. Particular ethical issues related to the study and ethical permission numbers from the University of Atatürk University are noted at the end of the article.

FINDINGS

Middle School Students' 21st Century Skills

When the pre- and post-intervention scores of the 21st Century Skills Scale were compared, the results indicated a positive effect of the implemented training or activities on the participants. Although participants' perceptions of their 21st century skills were relatively high before the intervention ($\bar{X}$ = 4.01, SD = 0.67), a noticeable increase was observed after the intervention ($\bar{X}$ = 4.57, SD = 0.57).

Table 3. Wilcoxon Test Findings Regarding 21st Century Skills

Post-Pre Test	N	Mean Rank	Sum of Ranks	Z	p
Negative ranks	1[a]	7.00	7.00	-2.513	0.012
Pozitive ranks	12[b]	6.45	71.00		
Ties	2[c]				

According to the 21st Century Skills Scale, a significant difference was found in favor of the post-test scores. Based on the effect size formula, the calculated r value was 0.67, indicating that the intervention had a strong positive impact on students' 21st century skills and was found to be highly effective.

Middle School Students' Self-Efficacy Perceptions of Computational Thinking Skills

As shown in Table 4, the comparison of pre- and post-intervention scores on the Computational Thinking Self-Efficacy Scale and its sub-dimensions indicates that the implemented training had a positive impact on the participants. In the data processing sub-dimension, participants' scores were at a low level before the intervention ($\bar{X}$ = 2.36, SD = 1.35), but a significant increase was observed afterward ($\bar{X}$ = 4.59, SD = 0.41). In the basic programming sub-dimension, the average score increased from ($\bar{X}$ = 3.17, SD = 1.20) before the intervention to ($\bar{X}$ = 4.45, SD = 0.61) after the intervention. For the self-confidence sub-dimension, scores rose from ($\bar{X}$ = 3.05, SD = 1.08) pre-intervention to ($\bar{X}$ = 4.35, SD = 0.61) post-intervention. In the algorithm sub-dimension, the average score increased from ($\bar{X}$ = 3.25, SD = 0.92) before the intervention to ($\bar{X}$ = 4.39, SD = 0.43) after the intervention. Finally, in the problem-solving sub-dimension, scores rose from ($\bar{X}$ = 3.02, SD = 0.86) to ($\bar{X}$ = 4.45, SD = 0.50) following the intervention.

Table 4. CT Self-Efficacy Scale Pre-Intervention and Post-Intervention Descriptive Analysis Results

	Pre Test		Post Test	
CT Self-Efficacy Scale and Subscales	$\bar{X}$	Sd	$\bar{X}$	Sd
Data processing	2.36	1.35	4.59	0.41
Basic programming	3.17	1.20	4.45	0.61
Self-confidence	3.05	1.08	4.35	0.61
Algorithm	3.25	0.92	4.39	0.43
Problem Solving	3.02	0.86	4.45	0.50

Table 5 shows that the intervention created a significant difference in the sub-dimensions of data processing [Z = −3.184, p = 0.001, r = 0.851], basic programming [Z = −2.589, p = 0.001, r = 0.692], self-confidence [Z = −2.366, p = 0.018, r = 0.632], algorithm [Z = −3.043, p = 0.002, r = 0.813], and problem-solving [Z = −3.181, p = 0.001, r = 0.850].

Table 5. Wilcoxon Test Findings for CT Skills

	Z	p	Effect Size (Z/√N)
Data processing	-3.184	0.001	0.851
Basic programming	-2.589	0.001	0.692
Self-confidence	-2.366	0.018	0.632
Algorithm	-3.043	0.002	0.813
Problem solving	-3.181	0.001	0.850
Total	-3.301	0.001	0.882

A particularly strong effect was observed in data processing (r = 0.851), algorithm (r = 0.813), and problem-solving (r = 0.850) skills, indicating that the intervention effectively enhanced participants' cognitive processes and problem-solving abilities. In the basic programming sub-dimension [Z = −2.589, p = 0.001, r = 0.692], a large effect size was identified, demonstrating a significant improvement in participants' algorithmic thinking and programming competencies. When the general situation is evaluated, it is seen that there is a high effect size [Z=-3.301,p=0.001,r=0.882] between the total scores pre and post tests and the implementation made a strong difference. While a strong effect was achieved especially in technical and cognitive skills, an effective but relatively lower improvement was observed in the self-confidence dimension compared to the others.

The Effect of Programming-Based Science Activities on Secondary School Students According to Educators

In Figure 5, the experiences of the educators during the two weeks of the summer course were divided into categories and subcategories by content analysis based on their daily reflective diaries and their responses to the interview questions at the end of the process.

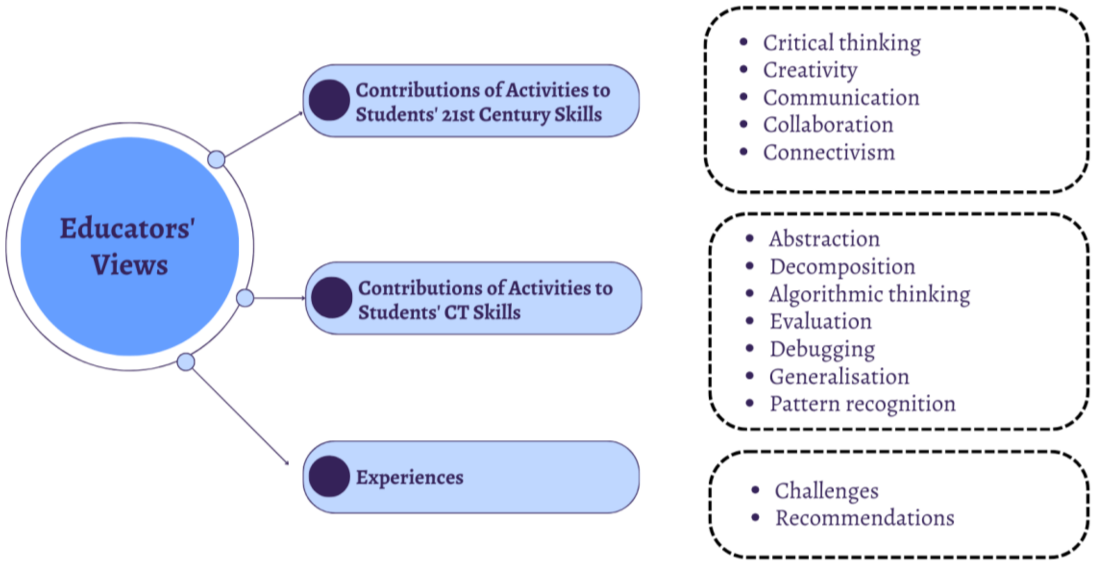

Figure 5. *Educators' Views*

Educators' Opinions on the Effect of Activities on Students' 21st Century Skills

At the end of the summer course, it was observed that the educators provided positive feedback on the development of students' 21st century skills. In the creativity dimension, it was emphasized that students developed original and innovative approaches to their project ideas. In terms of critical thinking, it was stated that students made joint decisions by discussing their ideas and produced alternative solutions by questioning. In communication and co-operation skills, it was observed that students established effective communication, harmonised in group work and developed joint projects. In the connectivity dimension, it was stated that students gained experience in accessing resources and collaborating by using digital tools for educational purposes. The quotations of the educators on the subject are shown in Table 6.

Table 6. Educators 'Views on Students' 21st Century Skills

21st Century Skill Dimensions	*Sample Quote*
Creativity	*E1:"Students' creativity skills are higher than mine, they think creatively, for example, their ideas when finding a project topic were very creative."*
	E2: "Both the students have different questions and they have different thoughts, they can produce ideas in a completely different way. Honestly, I like it very much both to improve myself and to see that children comprehend some things with such an education."
	E3:"I think creativity skills developed especially when we developed projects, each of them came up with very bright ideas. They created different projects, for example, they dreamed of providing the electricity of the football field from solar panels."
Critical Thinking	*E1: "In terms of critical thinking, they discussed their ideas with each other and made joint decisions."*
	E2: "At least in terms of a robot, when we ask what robots do and why we use them, they can establish that APoKS that we describe. When we go to space, they can reason that people may not survive, but robots can."
	E3: "I think our questions in the lesson plans we prepared helped them develop critical thinking skills because they questioned the examples I gave by asking what would happen if it were like this."
Communication &Collaboration	*E1:"Students, despite meeting for the first time, communicated and cooperated surprisingly well—listening to each other, discussing ideas, and making joint decisions while creating projects."*
	E2: "My group was actually more adolescent, but they co-operated well together, one of them wrote code and the other physically built the circuit, for example"
	E3: "There was no problem in co-operation, the students were not few in number, but they still listened to each other respectfully and a joint project came out of the group"
Connectivism	*E1: "For example, I set up a class on connectivity and they enrolled in Tinkercad. They could easily contact each other through the network and find exemplary projects."*
	E2: "Students actually have social media accounts, but they did not use them for education and enrol in classes, we did this in Tinkercad and created our projects, at least they had an experience."
	E3: "Now they know where to find what and how to find it, we showed them how we find educational content on the internet and I think they realised it a little more."

Educators 'Opinions on the Effect of Activities on Students' Computational Thinking Skills

They state that the educators observed the development of their students' algorithmic thinking skills after the summer course. In terms of decomposition, they state that although they had difficulties in the beginning, they have improved in time in terms of decomposing problems into their components and producing solutions. In terms of abstraction skills, they observed that they were able to identify possible problems by predicting certain scenarios, such as transitions between scenes in Scratch. In the evaluation process, they stated that their ability to analyse the codes, to notice deficiencies and to produce better solutions increased. In terms of debugging and error finding, they stated that both they and their students developed these skills by conducting one-to-one trials during Arduino applications. In terms of pattern recognition, they stated that students were able to distinguish structural similarities and deficiencies when they examined the codes. Finally, they state that generalisation skills have also improved because students are able to generalise over certain code structures and problems and adapt them to new situations. The quotations of the educators on the subject are shown in Table 7.

Table 7. Educators 'Views on Students' CT Skills

CT Skill Dimensions	*Sample Quote*
Algorithmic thinking	*E1: "Before using these applications, they had no knowledge about Scratch and Arduino, even the simplest subject, Hour of Code. But I observed that the applications clearly improved their algorithmic thinking skills."*
	E2: They knew the value of step-by-step problem-solving but hadn't applied it before. Through the activities, they gained hands-on experience and now approach problems more critically and independently."
	E3: "At the beginning, my students did not know exactly how to plan step by step when solving a problem. However, over time, when they saw a problem, they started to break it down into smaller steps and solve it systematically."
Decomposition	*E1:"Initially, students saw projects as a whole and felt unsure where to start. Later, they learned to break them into steps—like planning sensor functions before movement in a robot project."*
	E2: "As the process progressed, they realized that they could break down a big problem into smaller and more manageable steps."
	E3: At first, students saw complex problems as a whole, but over time, they learned to break them down. For example, when designing a game, they planned character movements before background changes."
Abstraction	*E1: "For example, while doing an LED lighting project on Arduino, they first grasp the basic logic and then apply this knowledge to other projects."*
	E2: "They understood what they learnt, but now they started to think about how to use it in different examples."
	E3: "With Scratch, they now have a better understanding of which variables and commands are required when switching between scenes. They can eliminate unnecessary details and focus only on the important components."
Evaluation	*E1: "There were so many different ideas in the project that I was very happy. Some of them had problems with the loops. We helped them as their teachers. We helped them try and evaluate different ways and find the most appropriate one."*
	E2: "They started to evaluate and make their solutions both individually and with the group. After one or two LEDs exploded in the materials, they tried to think of the safest and most appropriate solutions."
	E3: "When reviewing their code, they can recognize which parts are working more efficiently and where they can make improvements. They now discuss which solution is more effective when evaluating each other's projects."
Debugging	*E1: "They can now examine the code line by line and realize what is wrong in loops or conditional statements. For example, one of my students checked the connections and the code one by one and found the error himself when the motor didn't work."*
	E2:"For instance, we wanted our group's planets to rotate, but they were spinning too fast—Saturn in particular was moving erratically. While troubleshooting, one student said, 'I placed this code here, and when I said turn 15 degrees, it turned.' He calculated the degree himself, explaining, 'I didn't want the planet to rotate fully, so I set it to 15 degrees."
	E3: "At first, students struggled to identify errors in their code. Over time, they learned to troubleshoot and now quickly spot issues like connection errors or missing commands in Arduino setups."
Pattern recognition	*E1: "When they were doing different projects, they started to realize that the blocks of code used were similar. For example, they realized that the code to make a character move in Scratch and the code to rotate a motor in Arduino are logically similar."*
	E2:"Initially, they built code by trial and error, but over time, they began to understand how blocks worked and related to each other. By the end, they could predict the output of Scratch blocks, a skill that improved further after the summer course."
	E3: "Students can now recognize certain patterns when analyzing code. For example, they can see similarities between repeating loops and conditional statements and use them in new projects."
Generalisation	*E1:"They transfer what they've learned to new problems—for example, using loop logic from a Scratch game to control a 10-light Arduino circuit instead of the requested three. They now apply code structures across different platforms."*
	E2: "They were able to generalize our examples to other situations very quickly."
	E3: "They better understood how to apply what they learned in one project to another project. For example, they try to apply the logic they use when designing a game in Scratch to Arduino circuits."

In summary, when the common opinions of the educators regarding the summer course were analysed, they stated that all of their students were able to follow the visual and physical programming steps correctly and perform the given tasks completely. The implementations improved the algorithmic thinking skills of the students. It was observed that the students, who initially had limited knowledge about Scratch and Arduino, were able to produce their own solutions by better understanding the problem solving steps in the process. In terms of decomposing problems, it was stated that students first handled complex projects as a whole, but over time they made them more manageable by dividing them into smaller components. In terms of abstraction skills, it was stated that students were able to comprehend basic concepts and apply them in different projects, and focus on important components by eliminating unnecessary details. In the evaluation process, it was reported that students started to determine the most appropriate methods by analysing their solutions individually and in groups. It was emphasised that debugging skills improved and students were able to detect errors faster and produce solutions by examining the codes line by line. Within the scope of pattern recognition skills, it was stated that students started to recognise similar code blocks and were able to use these structures in different projects. Finally, it was stated that generalization skills improved, and students could easily adapt the concepts they learned in a project to different problem situations. At the end of the implementation, the educators thought that the lesson plans and activities they prepared kept the students' interest in the lesson alive and created a basis for new learning and questions. The fact that the activities they prepared worked correctly and enabled the students to achieve the targeted gains increased their self-confidence. It is seen that the perspective of the educators, who want to develop and use these skills in their future lives, has changed positively towards their professions and the course. According to the evaluations of the educators in the projects of the groups, concepts such as sound, force and energy were concretised in the projects of the groups. For example, the Civcivler group demonstrated how to use sound as a warning tool with the security alarm project. The Magnafen group integrated science knowledge into real life in the gas alarm project and had the opportunity to experience science in practice by using sensors that detect methane gas. Technological tools such as Arduino and sensors were used by the groups and integrated with science education. This process increased students' technological literacy and made them more competent in scientific experimentation and research. The Hababam group enabled students to develop innovative thinking skills by combining technology with science education through the design of a parking sensor.

Educators' Experiences: Challenges and Recommendations

Based on the observations of the educators during the summer course, their experiences regarding the difficulties and suggestions they encountered in practice were analysed. The quotations in Table 8 and Table 9 convey their experiences about the process.

Challenges

Among the difficulties encountered during the summer course from the educators‘ point of view, it was stated that some students’ lack of basic technology and programming knowledge caused them to have less time to produce more creative examples within 30 hours. It was observed that students needed more time to transition to advanced topics. The opinions of the educators on this issue are given in Table 8. During the summer course, the educators stated that time management was one of the biggest challenges. The limited time to complete the projects caused the students to rush through some stages and not focus enough on the details. In addition, technical problems experienced in the integration of physical and digital tools, especially when sensors did not work correctly, were among the factors that could negatively affect students' motivation. In addition, the lack of active participation of some students prevented the efficient completion of the projects by making intra-group co-operation difficult. It was observed that the individual working habits of the students were effective on group dynamics and this situation made co-operation difficult.

Table 8. Challenges in Summer Courses According to Educators

Use of Information Technologies	*E1: "Students needed more time to learn basic technology and programming concepts. Therefore, we could not focus enough on creative projects."*
	E2: "We had to spend more time in the beginning phase because some students did not know the basics well enough. This limited the time allocated for creative thinking during the project process."
	E3: "We had to spend time reinforcing basic skills before moving on to advanced projects. This allowed students to progress more efficiently in complex projects, but showed that more time was needed for the creative thinking process."
Time Management	*E1: "Since the projects had to be completed in the specified time, students had to pass some stages quickly and could not focus on the details sufficiently."*
	E2: "The time limit caused students to rush the thinking and problem solving stages. A more flexible time planning could have improved the quality of the projects."
	E3: "Students raced against time to complete certain steps in the project process. We observed that they needed additional time to analyze more deeply and develop original ideas."
Producing Creative Projects	*E1: "Students had difficulties in developing project ideas and often went for simple solutions. They could be helped to think creatively with more guidance."*
	E2: "Some students struggled due to the complexity of their projects. Taking smaller steps could have supported their creative thinking process."
	E3: "Students had difficulty in coming up with creative solutions to complete their projects. Guidance and sample applications could have facilitated this process."

In the creative thinking process, it was observed that students had difficulty in problem solving skills due to the complexity of the projects they determined. It was emphasised that such projects require more guidance, especially students need additional guidance to produce creative solutions. Although the educators thought that the determined project topics were sufficient to provide basic knowledge and skills, they stated that it would be useful to increase the time for the development of more original and creative projects.

Recommendations

Based on their experiences in the summer school, the educators shared their experiences and made suggestions in terms of concretisation of abstract concepts, development of problem solving and algorithmic thinking, interactive learning, creativity and innovative thinking, cooperation and teamwork, and attitude towards learning. The quotations in Table 9 include the suggestions for the use of visual and physical programming in science education based on the students' experiences during the summer course.

Table 9. Recommendations from Educators

Concreteization	*E1: "I think that visual and physical programming enables students to understand abstract concepts in science lessons in a more concrete way. In the future, such materials should be used more widely in the classroom."*
	E2: "Abstract science topics can be made concrete with more real-world applications. For example, in order to understand the technology around us, we can have students experience real-life devices with sensors and robots. This can help make abstract concepts more understandable."
	E3: "For example, by designing projects on current scientific topics such as sustainable energy sources and space exploration, we can help students understand these topics in more depth."
Problem Solving and Algorithmic Thinking Development	*E1: "I observed that when students write algorithms, their problem solving skills develop much faster. This practice should progress from easy to difficult in a way that supports both mathematical and scientific thinking skills of students in schools."*
	E2: "We can enable students to produce solutions to more complex problems with larger-scale group projects."
	E3: "To improve students' algorithmic thinking skills, we can design projects using more advanced programming languages. Students can test their algorithms on real world problems."
Interactive Learning	*E1: "These activities enabled my students to participate more actively in the lesson and learn the concepts more permanently. I plan to include more interactive applications in my lessons in the future."*
	E2: "For example, we can make science topics more engaging by using technologies such as augmented reality (AR) and virtual reality (VR). With such tools, students become a part of the lesson and participate more actively in learning."

	E3: "We can increase interactivity by using more digital learning tools in lessons. For example, by using online simulations and laboratories, we can allow students to perform science experiments in a virtual environment."
Creativity and Innovative Thinking	*E1: "Visual programming tools really improved students' creative thinking skills. They became more courageous in producing innovative solutions. I think such tools should be used more in the future."*
	E2: "By designing more open-ended projects, we can allow students to use their imagination."
	E3: "I think we have presented enough creative examples at this level with this time limit. Maybe organizing more advanced courses and longer projects would bring more creative ideas."
Collaboration and Teamwork	*E1: "Students learned a lot from each other during group work. In the future, I think teamwork-oriented projects should be increased."*
	E2: "We could exchange students between the teams or they could brainstorm and give ideas to each other in the process"
	E3: "Maybe in the future, we can bring students from different classes together and strengthen their problem solving skills with different perspectives."
Learning Attitude	*E1: "Their interest in science lessons has greatly increased. In the future, we should use more creative and interactive methods to keep students interested in these lessons."*
	E2: "We could have students take on roles like 'mini scientists' to make scientific discoveries."
	E3: "Maybe we could give students a research assignment and have them discuss the impact of past scientific discoveries on society..."

When the table is summarised, educators who use visual and physical programming applications in science education emphasise that this process provides significant contributions to students in terms of concretisation, problem solving, interactive learning, creativity, collaboration and positive learning attitudes. It was suggested to develop more storytelling and game-based content and to increase the number of interdisciplinary projects and applications for real-world problems. It is also suggested to add activities that encourage students to develop more original and innovative solutions by providing opportunities to design their own projects. In general, widespread use of programming in science education is effective for educators in creating a learning environment that strengthens students' 21st century skills.

DISCUSSION & CONCLUSION

In education, it is known that learning enriched with visual and physical programming activities has been frequently used in recent years to reduce students' misconceptions, increase their self-efficacy and gain permanent learning experiences due to the abstract subjects and high misconceptions of students (Güneş & Küçük, 2022). The topics that visual and physical programming activities generally focus on are examples such as change of state of matter (Karaşahin & Sarı, 2022), force and motion (Uçar & Sezek, 2024), and the use of wearable technologies in teaching electrical circuits (Nugent et al., 2019; Şat et al, 2025). In addition, it is seen that educational interventions are carried out to reduce learners' misconceptions and concretize abstract concepts in subjects such as showing the relationship between cell size and diffusion rate (Derman, 2023) or teaching with visual programming for the circulatory system (Aytekin & Topçu, 2025). However, in this study, it is thought to be important in terms of having an inclusive and integrative perspective in terms of developing and implementing activities by determining achievements for each unit by addressing the subjects in secondary school science education with a spiral program structure. Especially in the 2024 science curriculum, it was seen that activities with more context-based and experience-based approaches were targeted (Torun & Karamustafaoğlu, 2025). In this study, lesson plans and activities were prepared to support the 2024 Türkiye Century Maarif Model, and a complementary and complementary approach to the existing programs was adopted.

When the quantitative and qualitative findings of the study were interpreted together, it was seen that the science activities based on visual and physical programming carried out in the summer school made a significant difference on the 21st century thinking skills and CT self-efficacy perception. After analyzing the quantitative data, it was seen that there was a change in students' 21st century skills at a high impact level. In the CT self-efficacy perceptions, it is noteworthy that there is an improvement at a high impact level in all of the dimensions of algorithm designing, problem solving, data processing, basic programming and self-confidence in middle school students after the application. In the literature, it is known that programming activities improve CT and self-efficacy (Erümit et al., 2025).

When the findings from the structured interviews with the pre-service teachers and the analysis of the reflection diaries were analyzed, it was seen that the summer course students' views on 21st century skills were examined and it was seen that they thought that their problem solving, creativity, cooperation and communication skills were improved. In this study, it is seen that a comprehensive evaluation approach was adopted for the process as a result of evaluating the students' performances with scales as well as supporting them with teachers' opinions. While quantitative data allow for an objective analysis of students' CT skills and the development of 21st century competencies, qualitative data allow for an in-depth examination of students' experiences, thoughts and the transformation in their learning processes throughout the process. In the study, similar results were reached with the studies in which it was realized that students learned to distinguish problems and divide them into sub-steps and were able to create an algorithm using a flowchart in solving the problem (Atabay, 2019). Unlike other studies conducted in secondary school samples (Durmuş, 2024), an increase in learners' 21st century skills was observed. In addition, an increase was observed in critical thinking, problem solving (Koca, 2020) and computational thinking (Durmuş, 2024). It is important that the objectives are clear and measurable, especially in teaching programming and computational thinking skills to children (Liu et al., 2024). In this study, measurable goals were set in the lesson plans based on measurable goals with the activities based on the APoKS prepared in the lesson plans, and it was seen that the students achieved the targeted outcomes after the implementation. The coding errors encountered by the students in the projects they prepared both supported their problem-solving skills by developing their critical thinking skills and provided peer-supported development of debugging, patterning and abstraction skills in terms of computational thinking skills. When they integrated the topics they chose in the activities with programming, they went through a process based on life-based learning and context-based learning and actively used mathematics, engineering, technology and design skills in their projects by transferring their skills in an interdisciplinary way (Aytekin & Topçu, 2025; Yüksel et al, 2025).

The findings of the study showed that lesson plans prepared with a STEM approach and block-based programming activities based on APoKS and physical programming activities supported students' development of CT and 21st century skills. Similar to the literature, it is thought that robotics and coding education in science education provides development in areas such as technological adaptation, creative thinking, problem solving, digitalization and 21st century skills (Top & Arabacıoğlu, 2021; Büyük, 2024; Rapti & Sapounidis, 2024; Yüksel et al, 2025). However, differently, according to the opinion of the educators in this study, visual programming activities could be performed more easily than physical programming. The main reason for this can be thought to be that they need more information in visual programming related to mathematics and that interdisciplinary connections are used more in those activities (Bilgic & Doğusoy, 2023). In this context, enriching the lessons with block-based programming activities, providing students with active participation opportunities, addressing the course outcomes in a real dimension, and using gamified activities can motivate students to both lessons and programming (Bilgic & Doğusoy, 2023). Similarly, it has been emphasized that student-oriented approaches are important in designing educational processes involving information technologies and examining student participation in activities (Yıldız-Durak & Sarıtepeci, 2018; Sarıtepeci, 2020). In this study, it was observed that the visual and physical programming activities including APoKS for students were useful in programming education in a peer-supported manner. In line with the findings obtained, it was aimed to support students' data interpretation, analyzing data from different information sources, critical thinking in problem solving processes and alternative model development skills targeted by PISA, and it was observed that positive developments were made in this context. As reported by Aydın and Çilek (2024), it is known that a significant number of students have low satisfaction levels, face behaviors such as exclusion and discrimination, and have a low sense of belonging to the school, which may be the reason for the observed low achievement in international exams. Therefore, it is thought that the educational interventions used in this study will be a useful approach in preventing the inequalities of opportunity that students face in education and in overcoming the barriers to education and training processes such as peer bullying that they face at school because they are in a project-based learning environment with peers. Finally, these results show that the intervention was very effective in terms of improving the technical skills of the participants, but additional supportive strategies may be needed to further increase self-confidence.

Limitations & Further Research

- The educational intervention in this study—science activities based on visual and physical programming—was implemented over a 10-day (30-hour) summer school program with a single group. Therefore, future studies could adopt an experimental design with control groups to test students' skills more rigorously or extend the duration of the implementation to allow for a more comprehensive tracking of the process.

- Although many studies on science education and coding tend to suggest supplementary activities without altering the existing curriculum, it is recommended that future research develop and implement a comprehensive science education program grounded in visual and physical programming, followed by an evaluation of its impact.
- Based on the instructors' observations, it is suggested that future projects provide longer-term engagements and more free project time to better support students' creative thinking. This approach is believed to not only foster the development of technical skills but also encourage students to generate original ideas and innovative solutions. It was noted that the use of the 21st-century skill of *connectivism* remained limited in the planned activities. Future implementations could aim to integrate the use of social networks and social media in educational contexts to allow learners to become part of a research and inquiry-based community, thereby enhancing their Community of Inquiry (CoI) experiences.
- The study utilized *Hour of Code* and *Scratch 3.0* for visual programming, and Tinkercad with Arduino for physical programming. Future applications could incorporate diverse educational robotics kits, social robots, or AI-supported coding platforms to broaden the scope of technological engagement.
- The study was limited to voluntary students from public and private schools. Similar studies involving different age groups and individuals with special needs are recommended for future research.
- Activities involving AI-supported coding applications could be designed to enhance students' computational thinking skills and programming self-efficacy within instructional interventions aligned with Computer Science Pedagogical Content Knowledge.
- Finally, to shift the perception of schools from being merely transitional phases to essential parts of life itself, it is proposed that curriculum design move beyond overwhelming students with predefined outcomes. Instead, it should focus on real-life problem solving and interdisciplinary learning experiences, incorporating contemporary technological applications to better prepare students for the demands of the modern world.

Statements of Publication Ethics

In this study, the principles of publication ethics have been adhered to, and the ethical permission for the research has been approved by the Ethics Committee of Ataturk University Institute of Educational Sciences with the document number E-56785782-050.02.04-2200101348 on March 31, 2022.

Researchers' Contribution Rate

Authors	Literature review	Method	Data Collection	Data Analysis	Results	Conclusion
Author 1	☒	☒	☒	☒	☒	☒
Author 2	☒	☒	☐	☐	☒	☒

Conflict of Interest

The authors have no conflicts or competing interests to declare.

Acknowledgement

This article is derived from the first author's PhD dissertation completed in 2023 under the supervision of the second author.

REFERENCES

Alsancak, E., & Aybek, B. (2023). Ortaokul öğrencilerinin eleştirel düşünme eğilimleri ile akademik başarıları arasındaki ilişkinin incelenmesi. *Scientific Educational Studies, 7*(1), 57-78. https://doi.org/10.31798/ses.1289101

Amadi, C. S. (2022). The Integration of 21st-Century Skills in Science: A Case Study of Canada and the USA. Education and Urban Society, 55(1), 56-87. https://doi.org/10.1177/00131245211062531

Arslanhan, A., & Artun, H. (2021). Bilgi İşlemsel Düşünme Becerilerinin Fen Öğretimine Entegrasyonu Hakkında Öğretmen Görüşleri. *Eğitim Bilim Ve Araştırma Dergisi, 2*(2), 108-121.

Aşkun, V. (2024). Yapay Zekâ ve Otomasyon Çağında Eşitlik ve Refah: Daron Acemoğlu'nun Görüşlerine Dayalı Bir İnceleme. *Bozok Sosyal Bilimler Dergisi,* 3, 137-60.

Atabay, E. (2019). Okul öncesi dönem çocuklarına oyunlaştırma ile algoritma eğitimi verilmesi. [Yayımlanmamış yüksek lisans tezi].Süleyman Demirel Üniversitesi.

Aydın, A. T., & Çilek, A. (2024). PISA 2022 Sınav Sonuçları Bağlamında Türk Eğitim Sisteminin Genel Görünümü. *Abant İzzet Baysal Üniversitesi Eğitim Fakültesi Dergisi, 24*(4), 2361-2378. https://doi.org/10.17240/aibuefd.2024..-1550406

Aytekin, A., & Topçu, M. S. (2025). Teaching of the circulatory system subject with the plugged computational thinking-oriented module. Science Activities, 1–21. https://doi.org/10.1080/00368121.2025.2509952

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

Başaran, M. (2018). *The applicability of STEM approach in preschool education (action research)* (Unpublished doctorate thesis). Gaziantep University, Gaziantep.

Bers, M.U. Coding as another language: a pedagogical approach for teaching computer science in early childhood. *J. Comput. Educ.* **6**, 499–528 (2019). https://doi.org/10.1007/s40692-019-00147-3

Bilgic, K., Dogusoy, B. Exploring secondary school students' computational thinking experiences enriched with block-based programming activities: An action research. *Educ Inf Technol* **28**, 10359–10384 (2023). https://doi.org/10.1007/s10639-023-11583-1

Bozkurt, Ş. B., & Çakır, H. (2016). Ortaokul öğrencilerinin 21. yüzyıl öğrenme beceri düzeylerinin cinsiyet ve sınıf seviyesine göre incelenmesi. *Pamukkale Üniversitesi Eğitim Fakültesi Dergisi*, *39*(39), 69-82.

Bozkurt, A., Hamutoğlu, N.B., Liman Kaban, A., Taşçı, G. ve Aykul, M. (2021). Dijital bilgi çağı: Dijital toplum, dijital dönüşüm, dijital eğitim ve dijital yeterlilikler. *Açıköğretim Uygulamaları ve Araştırmaları Dergisi (AUAd), 7*(2)*,* 35-63. https://doi.org/10.51948/auad.911584

Bülegin, S., & İlkörücü, Ş. (2023). Eleştirel Düşünme Temelli Fen Etkinlikleri ile 6. Sınıf Öğrencilerinin Eleştirel Düşünme Becerilerini Geliştirmeye Yönelik Bir Eylem Araştırması. *Journal of Uludag University Faculty of Education, 36*(3), 1119-1146. https://doi.org/10.19171/uefad.1362486

Büyük, U., (2024). Eğitimde robotik ve kodlama uygulamaları: Fen Bilimleri öğretmenlerinin görüşleri. *Eğitimde Yeni Yaklaşımlar Dergisi (EYYAD), 7*(2), 96-117. https://doi.org/10.70325/eyyad.1487105

Bybee, R. W. (2013). The case for STEM education: Challenges and opportunities. NSTA press.

Cansu, F. K., & Cansu, S. K. (2019). An overview of computational thinking. *International Journal of Computer Science Education in Schools*, *3*(1), 17-30. https://doi.org/10.21585/ijcses.v3i1.53

Cantlon, J. F., Becker, K. T., & DeLong, C. M. (2024). Computational thinking during a short, authentic, interdisciplinary STEM experience for elementary students. *Journal for STEM Education Research*, 1-19. https://doi.org/10.1007/s41979-024-00117-0

Cheng, S., Ma, L., Lu, H., Lei, X., & Shi, Y. (2021). Evolutionary computation for solving search-based data analytics problems. *Artificial Intelligence Review*, *54*(2), 1321-1348. https://doi.org/10.1007/s10462-020-09882-x

Coxon, S. V. (2012). The malleability of spatial ability under treatment of a First Lego League based robotics simulation. *Journal for the Education of the Gifted, 35*(3), 291-316.

Creswell, J. W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches* (4nd ed.). Thousand Oaks, CA: Sage.

Çiftci, S., Yayla, A., & Sağlam, A. (2021). 21. yüzyıl becerileri bağlamında öğrenci, öğretmen ve eğitim ortamları. *RumeliDE Dil Ve Edebiyat Araştırmaları Dergisi(24)*, 718-734. https://doi.org/10.29000/rumelide.995863

Çorlu, M. S., Capraro, R. M., & Capraro, M. M. (2014). Introducing STEM education: Implications for educating our teachers in the age of innovation. Eğitim ve Bilim, 39(171), 74-85.

Çorlu, M. S., & Çallı, E. (2017). STEM kuram ve uygulamaları. *İstanbul: Pusula*, *20*.

Davenport, T. H. (2018). From analytics to artificial intelligence. *Journal of Business Analytics*, *1*(2), 73–80. https://doi.org/10.1080/2573234X.2018.1543535

Dede C. (2010). Comparing frameworks for 21st-century skills. *21st-century skills: Rethinking how students learn*. 20, 51–76.

Demir, D., & Seferoğlu, S. S. (2017). Yeni Kavramlar, farklı kullanımlar: Bilgi-işlemsel düşünmeyle ilgili bir değerlendirme. İçinde B. Akkoyunlu, A. İşman & H. F. Odabaşı (Ed). Eğitimde Teknoloji Okumaları (s.801-830). Ankara: TOJET.

Derman, M. (2023). Biyoloji Dersi İçin Arduino Tabanlı Deney Tasarımı. *Fen Bilimleri Öğretimi Dergisi, 11*(1), 180-204. https://doi.org/10.56423/fbod.1210918

Durmuş, R. (2024). Ortaokul öğrencilerine yönelik kodlama eğitimlerinin STEM ve bilgi işlemsel düşünme becerilerine etkisi. Yayınlanmamış yüksek lisans tezi. Balıkesir Üniversitesi Fen Bilimleri Enstitüsü.

Er, Z. (2024). Examination of the relationship between mathematical and critical thinking skills and academic achievement. *Pedagogical Research, 9*(1), em0176. https://doi.org/10.29333/pr/14028

Erumit, A. K., Ozmen, S., & Cebeci, H. Y. (2025). The impact of using various programming tools on pre-service teachers' self-efficacy beliefs and computational thinking skills. *The International Journal of Eurasia Social Sciences*, *16*(59), 80–97. https://doi.org/10.70736/ijoess.540

Eteng, I., Akpotuzor, S., Akinola, S. O., & Agbonlahor, I. (2022). A review on effective approach to teaching computer programming to undergraduates in developing countries. *Scientific African*, *16*, https://doi.org/10.1016/j.sciaf.2022.e01240

Facione, P. (1990). *Critical thinking: A statement of expert consensus for purposes of educational assessmentand instruction. Executive Summary: The Delphi Report*. California Academic Press.

Falloon, G. (2019). Using simulations to teach young students science concepts: An experiential learning theoretical analysis. *Computers & Education, 135*, 138–159.

Fırat, M., Kabakçı Yurdakul, I., & Ersoy, A. (2014). Bir eğitim teknolojisi araştırmasına dayalı olarak karma yöntem araştırması deneyimi. *Journal of Qualitative Research in Education, 2*(1), 65-86. [Online]: www.enadonline.com, doi: 10.14689/issn.2148-2624.1.2s3m

Gezer, M., & Durdu, L. (2025). An investigation of technology-rich lesson plans: Science teachers' views on technology integration. *Participatory Educational Research, 12*(1), 264-286. https://doi.org/10.17275/per.25.14.12.1

Gretter, S., Yadav, A. Computational Thinking and Media & Information Literacy: An Integrated Approach to Teaching Twenty-First Century Skills. *TechTrends* (60), 510–516 (2016). https://doi.org/10.1007/s11528-016-0098-4

Gunes, H & Kucuk, S. (2022). A systematic review of educational robotics studies for the period 2010–2021. *Review of education*, *10*(3), e3381. https://doi.org/10.1002/rev3.3381

Gülbahar, Y. , Kert, S. B. & Kalelioğlu, F. (2019). Bilgi İşlemsel Düşünme Becerisine Yönelik Öz Yeterlik Algısı Ölçeği: Geçerlik ve Güvenirlik Çalışması. Turkish Journal of Computer and Mathematics Education (TURCOMAT) , 10 (1) , 1-29. DOI: 10.16949/turkbilmat.385097

Güllü Egin, E., & Sözer, M. A. (2024). Bilgi işlemsel düşünme kavramının gelişim süreci. Bayburt Eğitim Fakültesi Dergisi, 19(44), 3101-3125

Gümüş, B., & Eroğlu, E. (2024). Fen eğitiminde gerçekleştirilen STEM uygulamalarının farklı değişkenler açısından incelenmesi: Bir meta analiz çalışması, *Trakya Eğitim Dergisi, 14*(1), 54-74. https://doi.org/10.24315/tred1259577

Güven, E. ve Sülün, Y. (2023). Ortaokul 5.sınıf fen öğretiminde arduino destekli robotik kodlama etkinliklerinin kullanılması. *Erzincan Üniversitesi Eğitim Fakültesi Dergisi,25*(2), 225-236. https://doi.org/10.17556/erziefd.1116283

Hangün, M. E., Kalınkara, Y., Bayer, H. ve Tekin, A., (2022). Eğitimde robotik kullanımına yönelik araştırmaların incelenmesi: Bir içerik analizi çalışması. *Erzincan Üniversitesi Eğitim Fakültesi Dergisi, 24*(3), 558-578. https://doi.org/10.17556/erziefd.944933

Herdem, K., & Ünal, İ. (2018). STEM Eğitimi Üzerine Yapılan Çalışmaların Analizi: Bir Meta-Sentez Çalışması. *Marmara Üniversitesi Atatürk Eğitim Fakültesi Eğitim Bilimleri Dergisi, 48*(48). https://doi.org/10.15285/maruaebd.345486

Hoeg, D. G., & Bencze, J. L. (2017). Values underpinning STEM education in USA: An analysis of the next generation science standards. *Science Education, 101*(92), 278–301.

Honey, M., Pearson, G., Schweingruber, H., & (Eds.). (2014). *STEM integration in K–12 education: Status, prospects, and an agenda for research*. National Academies Press. https://doi.org/10.17226/18612

ISTE(2015). CT Leadership toolkit. http://www.iste.org/docs/ct-documents/ctleadershipt-toolkit.pdf?sfvrsn=4.

ISTE(2023). Explore the ISTE computational thinking competencies. https://www.iste.org/standards/iste-standards-for-computational-thinking

John, O. P., & Srivastava, S. (1999). The Big-Five trait taxonomy: History, measurement, and theoretical perspectives. In L. A. Pervin & O. P. John (Eds.), *Handbook of personality: Theory and research* (2nd ed., pp. 102–138). Guilford Press.

Julià, C., & Antolí, J. Ò. (2016). Spatial ability learning through educational robotics. *International Journal of Technology and Design Education, 26*(2), 185-203. https://doi.org/10.1007/s10798-015-9307-2

Kanaki, K., & Kalogiannakis, M. (2022). Assessing Algorithmic Thinking Skills in Relation to Age in Early Childhood STEM Education. *Education Sciences*, *12*(6), 380. https://doi.org/10.3390/educsci12060380

Karaahmetoğlu, K. (2019). Proje tabanlı Arduino eğitsel robot uygulamalarının öğrencilerin bilgisayarca düşünme becerileri ve temel STEM beceri düzeyleri algılarına etkisi [Yayımlanmamış yüksek lisans tezi]. Amasya Üniversitesi.

Karakaya, F., & Avgın, S.S. (2016). Effect of demographic features to middle school students' attitude towards FeTeMM (STEM). *Journal of Human Sciences, 13(*3), 4188-4198. doi:10.14687/jhs.v13i3.4104

Karaşahin, A., & Sarı, U. (2022). Maddenin Hal Değişimini İncelemek İçin Arduino Deneyi. *Fen Bilimleri Öğretimi Dergisi, 10*(1), 208-226. https://doi.org/10.56423/fbod.1108976

Kardeş, A. (2020). Robotik kodlama öğretimine yönelik zenginleştirilmiş e-kitap geliştirilmesi [Yayımlanmamış yüksek lisans tezi]. Afyon Kocatepe Üniversitesi.

Kazakoff, E. R., Sullivan, A., & Bers, M. U. (2013). The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood. *Early Childhood Education Journal*, *41*, 245-255. https://doi.org/10.1007/s10643-012-0554-5

Kazez, H., & Genc, Z. (2016). Research trends in Lego and robotic usage in education: A document analysis. Journal of Instructional Technologies & Teacher Education, 5(1).

Kelleher, C., & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM computing surveys (CSUR)*, *37*(2), 83-137.

Ketelhut, D. J., Mills, K., Hestness, E., Cabrera, L., Plane, J., & McGinnis, J. R. (2020). Teacher change following a professional development experience in integrating computational thinking into elementary science. *Journal of science education and technology*, *29*, 174-188. https://doi.org/10.1007/s10956-019-09798-4

Kert, S. B., Erkoç, M. F., & Yeni, S. (2020). The effect of robotics on six graders' academic achievement, computational thinking skills and conceptual knowledge levels. *Thinking Skills and Creativity*, *38*, 100714. https://doi.org/10.1016/j.tsc.2020.100714

Koca, E. (2018). STEM yaklaşımı ile basınç konusunda bir öğretim modülünün geliştirilmesi ve uygulanabilirliğinin incelenmesi (Yöktez No. 527261) [Yüksek Lisans Tezi, Aksaray Üniversitesi].

Koca, S. (2020). Investigation of the effect of educational robot applications on students' cognitive outcomes towards coding [Unpublished master thesis]. Amasya University.

Korkmaz, Ö., Çakır, R., & Özden, M. Y. (2017). A validity and reliability study of the Computational Thinking Scales (CTS). Computers in Human Behaviours, 72, 558- 569.

Kurudayıoğlu, M., & Soysal, T. (2018). 2018 Türkçe dersi öğretim programi'nin dijital yetkinlik bakimindan incelenmesi. *Mehmet Akif Ersoy Üniversitesi Eğitim Fakültesi Dergisi*, (54), 184-199.

Küçükaydın, M.A., Çite, H. & Ulum, H. (2024). Modelling the relationships between STEM learning attitude, computational thinking, and 21st century skills in primary school. *Educ Inf Technol* 29, 16641–16659 https://doi.org/10.1007/s10639-024-12492-7

Li, X., Xiang, H., Zhou, X., & Jing, H. (2023, June). An empirical study on designing STEM+ AI teaching to cultivate primary school students' computational thinking perspective. In *Proceedings of the 2023 8th International Conference on Distance Education and Learning* (pp. 1-7). https://doi.org/10.1145/3606094.3606130

Liu, Z., Gearty, Z., Richard, E., Orrill, C. H., Kayumova, S., & Balasubramanian, R. (2024). Bringing computational thinking into classrooms: a systematic review on supporting teachers in integrating computational thinking into K-12 classrooms. *International Journal of STEM Education*, *11*(1), 51.

Macrides, E., Miliou, O., & Angeli, C. (2022). Programming in early childhood education: A systematic review. *International Journal of Child-Computer Interaction*, *32*, 100396. https://doi.org/10.1016/j.ijcci.2021.100396

Mazzola-Randles, C. (2020). To what extent does education prepare students for the 21st century to become self-regulated, future-proofed students, using networked, technological environments. https://doi.org/10.13140/RG.2.2.18400.10246

Mete, G. (2021). Ortaokul Öğrencilerinin Eleştirel Düşünme Becerilerinin İncelenmesi. *Ana Dili Eğitimi Dergisi, 9*(2), 492-509. https://doi.org/10.16916/aded.887592

Mezirow, J. (1996). Contemporary paradigms of learning. *Adult education quarterly, 46*(3), 158-172.

Nugent, G., Barker, B., & Grandgenett, N. (2008). *The effect of 4-H robotics and geospatial technologies on science, technology, engineering, and mathematics learning andattitudes*. In J. Luca, & E. Weippl (Eds.), Proceedings of world conference on educational multimedia, hypermedia and telecommunications (pp. 447–452). Chesapeake, VA:AACE.

Nugent, G., Barker, B., Lester, H., Grandgenett, N., & Valentine, D. (2019). Wearable textiles to support student STEM learning and attitudes. *Journal of Science Education and Technology, 28*(5), 470–479. https://doi.org/10.1007/s10956-019-09779-7

Ogegbo, A. A., & Ramnarain, U. (2022). A systematic review of computational thinking in science classrooms. *Studies in Science Education*, *58*(2), 203-230. https://doi.org/10.1080/03057267.2021.1963580

Özel, O. (2019). Effects of different programming methods on perception of self-efficacy for elementary students' computational thinking skills and programming achievement [Unpublished master thesis]. Marmara University.

Pallant, J. (2007). *SPSS Survival Manual: A Step By Step Guide to Data Analysis Using SPSS for Windows* (3rd ed.). Open University Press.

Partnership for 21st Century Learning P21 (2019). Framework for 21st century learning definitions. https://static.battelleforkids.org/documents/p21/p21_framework_brief.pdf adresinden 23.01.2025 tarihinde erişilmiştir.

Pellas, N. (2024). Enhancing Computational Thinking, Spatial Reasoning, and Executive Function Skills: The Impact of Tangible Programming Tools in Early Childhood and Across Different Learner Stages. *Journal of Educational Computing Research*, *63*(1), 3-32. https://doi.org/10.1177/07356331241292767

Perkovic, L., & Settle, A. (2010). Computational thinking across the curriculum: A conceptual framework. *College of computing and digital media technical report*, 10-001.

Polat, E., Hopcan, S., Kucuk, S., & Sisman, B. (2021). A comprehensive assessment of secondary school students' computational thinking skills. *British Journal of Educational Technology*, *52*(5), 1965-1980.

Polatgil, M. (2020). El-Cezerî ve çalışmalarının STEM eğitiminde kullanılması: deneysel bir çalışma. *Ağrı İbrahim Çeçen Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, 6*(2), 417-432. https://doi.org/10.31463/aicusbed.731166

Rapti, S., & Sapounidis, T. (2024). "Critical thinking, Communication, Collaboration, Creativity in kindergarten with Educational Robotics": A scoping review (2012–2023). *Computers & Education*, *210*, 104968. https://doi.org/10.1016/j.compedu.2023.104968

Román-González, M., Moreno-León, J., Robles, G. (2019). Combining Assessment Tools for a Comprehensive Evaluation of Computational Thinking Interventions. In: Kong, SC., Abelson, H. (eds) Computational Thinking Education. Springer, Singapore. https://doi.org/10.1007/978-981-13-6528-7_6

Sarı, U., & Karaşahin, A. (2020). Fen eğitiminde bilgi işlemsel düşünme: bir öğretim etkinliğinin değerlendirilmesi. *Turkish Journal of Primary Education*, *5*(2), 194-218.

Saritepeci, M. (2020). Developing computational thinking skills of high school students: design-based learning activities and programming tasks. *Asia-Pacific Education Researcher, 29*(1), 35–54. https://doi.org/10.1007/s40299-019-00480-2

Schleicher, A. (2018), *World Class: How to Build a 21st-Century School System*, Strong Performers and Successful Reformers in Education, OECD Publishing, Paris, https://doi.org/10.1787/9789264300002-en.

Selcuk, N.A., Kucuk, S. & Sisman, B. Does really educational robotics improve secondary school students' course motivation, achievement and attitude?. *Educ Inf Technol* **29**, 23753–23780 (2024). https://doi.org/10.1007/s10639-024-12773-1

Seraj, M., Katterfeldt, E. S., Bub, K., Autexier, S., & Drechsler, R. (2019). Scratch and Google Blockly: How girls' programming skills and attitudes are influenced. In *Proceedings of the 19th Koli Calling International Conference on Computing Education Research* (pp. 1-10).

Shang, X., Jiang, Z., Chiang, F., Zhang, Y., & Zhu, D. (2023). Effects of robotics STEM camps on rural elementary students' self-efficacy and computational thinking. *Educational technology research and development*, 1-26.https://doi.org/10.1007/s11423-023-10191-7

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational research review*, *22*, 142-158. https://doi.org/10.1016/j.edurev.2017.09.003

Soypak, B., & Eskici, M. (2023). Lise-Ortaokul Matematik, Fen Derslerinde Robotik Kodlama Uygulamalarına Yönelik Araştırmaların İncelenmesi: Bir İçerik Analizi Çalışması. *Fen Matematik Girişimcilik ve Teknoloji Eğitimi Dergisi, 6*(3), 214-229.

Sublette, H. (2013). An effective model of developing teacher leaders in STEM education [Unpublished doctoral dissertation].Pepperdine University.

Şat, M., Yardımcı, Ş. & Canbazoğlu Bilici, S. (2025). Investigating the Effects of E-Textiles on Students' Performance, Attitudes Toward Electric Circuits, and STEAM Attitudes: A Focus on Underrepresented Minorities. *J Sci Educ Technol*. https://doi.org/10.1007/s10956-025-10221-4

Şişman, B., & Küçük, S. (2019). An Educational Robotics Course: Examination of Educational Potentials and Pre-Service Teachers' Experiences. *International Journal of Research in Education and Science*, *5*(2), 510-531.

Şişman, B., Küçük, S., & Yaman, Y. (2021). *The Effects of Robotics Training on Children's Spatial Ability and Attitude Toward STEM. International Journal of Social Robotics.* doi:10.1007/s12369-020-00646-9

Topu, F., Baydaş, Ö., Turan, Z., Göktaş, Y. (2014). Common reliability and validity strategies in instructional technology research. *Çukurova Üniversitesi Eğitim Fakültesi Dergisi, 42*(1), 110-126. https://doi.org/10.14812/cuefd.54285

Top, O., & Arabacıoğlu, T. (2021). Bilgi işlemsel düşünme: Bir sistematik alanyazın taraması. *Uludağ Üniversitesi Eğitim Fakültesi Dergisi*, *34*(2), 527-567.

Torun, B., & Karamustafaoğlu, O. (2025). 2024 Türkiye Yüzyılı Maarif Modeli ile 2018 Fen Bilimleri Dersi Öğretim Programlarının Temel Öğeler Bakımından Karşılaştırılması. *Trakya Eğitim Dergisi, 15*(Özel Sayı), 346-388. https://doi.org/10.24315/tred.1614676

Tosik-Gün, E., & Güyer, T. (2019). Bilgi işlemsel düşünme becerisinin değerlendirilmesine ilişkin sistematik alanyazın taraması. *Ahmet Keleşoğlu Eğitim Fakültesi Dergisi*, *1*(2), 99-120.

Uçar, A. ve Sezek, F. (2024). Fen öğretiminde lego robotik uygulamalarının akademik başarıya etkisi. *Bolu Abant İzzet Baysal Üniversitesi Eğitim Fakültesi Dergisi, 24*(2) 585-510, https://dx.doi.org/10.17240/aibuefd.2024..-1311405

Uğraş, M., Güneş, H., Uğraş, H., Papadakis, S., & Kalogiannakis, M. (2025). Coding activities in early childhood: a systematic review. *International Journal of Technology Enhanced Learning*, *17*(1), 45-80. https://doi.org/10.1504/IJTEL.2025.143512

Voogt, J., & Roblin, N. P. (2012). A comparative analysis of international frameworks for 21st century competences: Implications for national curriculum policies. *Journal of curriculum studies, 44*(3), 299-321. https://doi.org/10.1080/00220272.2012.668938

Wang, C., Shen, J. & Chao, J. (2022). Integrating Computational Thinking in STEM Education: A Literature Review. *Int J of Sci and Math Educ* **20**, 1949–1972. https://doi.org/10.1007/s10763-021-10227-5

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2014). *Defining computational thinking for science, technology, engineering, and math*. Paper presented at the 2014

Annual Meeting of the American Educational Research Association, Philadelphia, Pennsylvania. https://ccl.northwestern.edu/papers/2014/CT-STEM_AERA_2014.pdf.

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

Xu, W., Geng, F. and Wang, L. (2022) 'Relations of computational thinking to reasoning ability and creative thinking in young children: mediating role of arithmetic fluency', *Thinking Skills and Creativity*, April, Vol. 44, p.101041, DOI: 10.1016/j.tsc.2022.101041.

Yalap, H. ve Gazioğlu, M. (2023). Türk eğitim sisteminde dijitalleşmeye yönelik uygulamalar. Manas Sosyal Araştırmalar Dergisi, 12(ÖS), 80-90. doi:10.33206/mjss.1301070

Yavuz Aksakal, N. & Ülgen, B. (2021). Yapay Zekâ ve Geleceğin Meslekleri . *TRT Akademi , 6* (13) , 834-853 . DOI: 10.37679/trta.969285

Yıldız-Durak, H & Saritepeci, M. (2018). Analysis of the relation between computational thinking skills and various variables with the structural equation model. *Computers & Education*, *116*, 191-202. https://doi.org/10.1016/j.compedu.2017.09.004

Yıldız-Durak, H., Yilmaz, F. G. K., & Yilmaz, R. (2019). Computational thinking, programming self-efficacy, problem solving and experiences in the programming process conducted with robotic activities. *Contemporary Educational Technology*, *10*(2), 173-197. https://doi.org/10.30935/cet.554493

Yılmaz, C. (2019). The effect of STEM education on 10th grade students' academic success, their attitude towards STEM and physics [Unpublished master thesis]. Pamukkale University.

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

Yüksel, A. O., Atasoy, B., & Özdemir, S. (2025). Transformation of labor: Educational robotics coding in elementary schools for 21st century skills. *Entertainment Computing*, *52*, 100890. https://doi.org/10.1016/j.entcom.2024.100890

Zakwandi, R., Istiyono, E. (2023). A framework for assessing computational thinking skills in the physics classroom: study on cognitive test development. *SN Soc Sci* 3, 46. https://doi.org/10.1007/s43545-023-00633-7