Research Article

PSYCHO-EDUCATIONAL RESEARCH REVIEWS

Psycho-Educational Research Reviews
11(*3*), 2022, 656-674
www.perrjournal.com

The Effects of Instructional Environments and Cognitive Abilities on Abstraction Performance

Fulya Torun, Assist. Prof. Dr., Aydın Adnan Menderes University, Dep. of Computer Education and Instructional Technologies, fulya.torun@adu.edu.tr 0000-0001-6942-888X

Arif Altun, Prof. Dr., Hacettepe University, Dep. of Computer Education and Instructional Technologies, altunar@hacettepe.edu.tr 0000-0003-4060-6157

Keywords

Computer science education
Teaching abstraction
Puzzle based learning
Working memory
Logical reasoning

Article Info:

Received : 21-04-2022
Accepted : 20-10-2022
Published : 10-12-2022

DOI: 10.52963/PERR_Biruni_V11.N3.18

Abstract

Abstraction is one of the building blocks in computer science (CS) and is described as omitting details and focusing on the necessary information. One of the approaches in teaching CS is puzzle based learning (PBL) approach which models problem solving process. Individual differences, on the other hand, exist as a mere fact in learning. Therefore, when designing instructional materials, it is essential to understand the interaction between individual differences along with the teaching paradigms. The first aim of this research is to investigate how students' working memory capacities (WMCs) and different learning environments based on puzzle based learning affect students' abstraction performance. 2X2 factorial design was utilized in the study. The second aim of the study was to investigate whether students' logical reasoning capacities (LRCs) and abstraction ability capacities (AACs), in each learning environments, had an effect on students' abstraction performances when their WMCs were controlled. According to the results of the research it was found that students' gender, abstraction skills and the learning environments had no effect on students' learning performances. In the other hand; the students with higher working memory capacities versus the students with low working memory capacities; the students with higher and medium logical reasoning level versus the students with lower logical reasoning level were found to have significantly higher learning performance. Also it was seen that logical reasoning levels of the students had predicted the learning performance but working memory performances of the students had not.

To cite this article: Torun, F., & Altun, A. (2022). The effects of ınstructional environments and cognitive abilities on abstraction performance. *Psycho-Educational Research Reviews, 11*(3), 656-674. doi: 10.52963/PERR_Biruni_V11.N3.18

INTRODUCTION

Abstraction is a high level cognitive process and an individual effort (Duncan, 2013; Saitta & Zucker, 2001; Sigel, 1953) which help us simplify the problems (Michalewicz et al., 2011), and reduce or exclude the unnecessary details (Falkner et al., 2012b; Forero et al., 2011; Guarino, 1978). When abstracting, one needs to focus (Statter & Armoni, 2017) and orient their attention to the content itself (Duncan, 2013), apply reasoning (Duncan, 2013; Kramer, 2007), think critically (Kuloğlu & Asasoğlu, 2010), apply generalization (Hill et al., 2008), and utilize the contexts (Hazzan & Kramer, 2007) in order to reduce cognitive complexity (Hazzan & Tomayko, 2005). Based on these characteristics, abstraction is a cognitive process where an individual conceptualizes a problem; use their reasoning skills when simplifying the related facts; utilize generalizations when extracting the irrelevant details among complex entities; and, use their cognitive effort optimally by directing attention to what is important.

Abstraction is one of the important and fundamental learning components in computer science (CS) curriculum (Bennedsen & Caspersen, 2008; Haberman & Muller, 2008; Hazzan & Tomayko, 2005). It is a complicated concept especially for middle school students to grasp (e.g. Armoni, 2013; Haberman & Muller, 2008); yet, it is an essential skill to be developed since it is used to make the problems easier to solve, such as the reduction of computational effort on problem solving (Zucker, 2003). Kramer (2007) stated that creating, designing, and implementing appropriate models for a particular purpose are essential for abstraction.

Research in teaching abstraction mainly focused on undergraduate students' abstraction abilities on different abstraction components (e.g. Hill et al., 2008), exploration the relationship between students' abstraction ability and programming performances (e.g. Bennedsen & Caspersen, 2006; Bennedsen & Caspersen, 2008), investigating the relationship between students' gender and teaching strategies (e.g. Statter & Armoni, 2017), and discussions about the importance of abstraction for CS teaching (e.g. Kramer, 2007). Altun (2012) claimed that due to the differences in cognitive and non-cognitive characteristics of the individuals, the learning processes also differ; therefore, the same learning outcomes may not occur at the same level for all in a uniformly designed learning environments. When individual differences are considered, it might be a necessity to provide multiple designs different learning environments taking these differences into consideration.

ABSTRACTION IN LEARNING AND TEACHING CS

Many researchers agree on that abstraction is an important component of CS (e.g. Armoni, 2013; Bennedsen & Caspersen, 2006; Colburn & Shute, 2007; Rich et al., 2019; Zehetmeier et al., 2019); frequently used in all processes in the field of CS, a strong cognitive ability for resolving complexity in programming; and, one of the important conditions for writing a good program (e.g. Haberman & Muller, 2008; Kramer, 2007). Bennedsen and Caspersen (2006) further added that abstract thinking is an important skill when learning CS concepts in general and programming in particular.

When solving a problem, software developers apply abstraction layers (e.g. Sprague & Schahczenski, 2002) and facilitate various levels of data and procedural abstractions (e.g. Colburn & Shute, 2007) in their programming tasks, where they further create functional abstractions (Sprague & Schahczenski, 2002). Colburn and Shute (2007) list some of the abstraction tasks utilized when programming;

- Deciding on data types like integer, longword, float and double where details are hidden about the representation of the numerical values of bytes
- Utilizing the variables, that hides the details of data values of symbolic tokens such as a location of data values in the memory
- Applying logical decisions in designing flip-flops

- Understanding and using registers, that hide the details of how bits are related with processor and memory
- Writing machine instruction, that hides the details of how bits represent processor operations on registers and variables

Although abstraction is included as a fundamental concept in CS textbooks; it is also hard to teach since its teaching steps are difficult to define (Rich et al., 2019; Zehetmeier et al., 2019). Learning the abstraction procedures and transferring these procedures in their software projects is a challenge for novice learners. They encounter abstraction as a concept or an approach in different contexts (Haberman & Muller, 2008). Researchers and educators therefore suggest abstraction to be included in their curriculum (Bennedsen & Caspersen, 2008; Haberman & Muller, 2008), maybe with a more direct focus on it (Kramer, 2007). In this manner choosing an appropriate learning approach, while teaching abstraction, is also important. Many studies show that puzzle based learning (PBL) approach is effective in learning and teaching CS (see Kawash, 2012; Merrick, 2010; Oyelere et al., 2019). But there is no study found that investigated this approach on learning and teaching abstraction CS. Therefore, it is seen to be a necessity to handle this approach in this research.

RESEARCH PROBLEM

Research about teaching abstraction in CS were investigated mostly with university students (Bennedsen & Caspersen, 2006; Bennedsen & Caspersen, 2008; Hazzan & Kramer, 2007), followed by high school students (Sakhnini & Hazzan, 2008; Taub et al., 2014), middle school students (Statter & Armoni, 2017; Statter & Armoni, 2020), and elementary school students (Çakıroğlu et al., 2021; Statter & Armoni, 2017). Abstraction has been explored mostly in programming courses (Çakıroğlu et al., 2021; Bennedsen & Caspersen, 2006; Bennedsen & Caspersen, 2008; Waite et al., 2018). In addition, some researchers have investigated abstraction in CS (Hazzan, 2003; Hill et al., 2008; Statter & Armoni, 2017), artificial intelligence (Saribatur et al., 2021), mathematics (Çetin & Dubinsky, 2017; Hazzan, 2003), and physics (Taub et al., 2014) courses. The variables in those studies were abstraction presentation (Wolz & Conjura, 1994), reducing abstraction (Hazzan, 2003; Sakhnini & Hazzan, 2008), abstraction ability (Bennedsen & Caspersen, 2006; Bennedsen & Caspersen, 2008), components of abstraction (Hill et al., 2008), abstraction as a process (Hazzan & Kramer, 2007), abstraction levels (Taub et al., 2014; Waite et al., 2018), abstraction in computational thinking (Çetin & Dubinsky, 2017), and teaching abstraction (Statter & Armoni, 2017). Bennedsen and Caspersen (2008) found that abstraction ability had an effect on students' performances in their other courses, as well. Taub et al., (2014), for example, reported that learning abstraction in CS also contributed to students' physics course performances.

Abstraction is considered as an element in the context of computer science, as well as a process in the cognitive context. During abstraction, many thinking processes work in cognition (Mirolo, 2022). In this manner one of the important ability for the process of abstraction is reasoning (Duncan, 2013; Kramer, 2007). Reasoning includes problem-solving, decision-making, learning etc. These all are concerned with computer science education. In this manner, there can be a relation with abstraction performances. So that, in Costa and Miranda's (2019) research it can be seen that logical reasoning capacities both can be affected by the learning process of programming or can affect to their learning process of programming. When the literature is reviewed, there are also different variables that can be thought to be effective in abstraction performance. As a cognitive process, abstraction should be done with focusing by reducing the unnecessary details (Statter & Armoni, 2017; Falkner et al., 2012b; Forero et al., 2011; Guarino, 1978). In this manner, the individuals' working memory capacity (WMC) is also important. Because WMC provides storing and manupilating the necessary information in some complex cognitive processes such as comprehension, learning, and reasoning (Baddeley, 1992). These three are also important in learning CS. For instance in the research of Bergersen and Gustafsson (2011), results indicated that the working memory has an effect on programming skills through programming knowledge. The results shows that there is a need for different variables that need to be worked with working memory.

To conclude, the results of those studies conducted in teaching abstraction showed that (a) learning performances could have been improved significantly by instructional methods (Wolz & Conjura, 1994), (b) instructional design could support students' learning (Waite et al., 2018), (c) girls' learning performances were better than boys, and girls' motivation increases with instructional guidance (Statter & Armoni, 2017). Yet, research is scarce to explore how instructional design choices might influence students' abstraction performances when their cognitive individual differences are taken into account, such as learners' working memory (WM) and logical reasoning capacities. At the same time when designing instruction, educators are urged to consider not only the individual differences, but also the weaknesses of learning designs (Salleh & Zakaria, 2011). According to Cowan (2014), these weaknesses could be overcome simply by applying the following three suggestions:

a. WM level of learners can be taken into account for teaching.
b. Training exercises can be taken by learners to improve their WM levels.
c. Considering the role of WM at a larger scale and its effects on the important goals of education.

Although these strategies have been put forward, no substantial research has been reported to show how learners' varying WM performances play a role when learning abstraction. Secondly, learners might possess different levels of WM capacities (WMCs), and they apply these cognitive resources at their best when exposed to certain abstraction task. Therefore, it is the first aim of this study to quest whether learners' WMCs could have any main or interaction effect in differently designed instructional environments when performing certain abstraction tasks. Secondly, it is investigated that whether students' logical reasoning capacities and abstraction abilities have any effect on their abstraction performances in each learning environments, when their WMCs are controlled.

In this study, puzzle based learning approach is focused on designing the learning environments; thanks to its contribution on learning and teaching CS. According to this, two different instructional environments have been utilized based on puzzle based approach to investigate each environments' effects on students' abstraction performances. These are content dependent instructional environment (CDLE) and content independent learning environment (CILE). In content independent learning environments, learners are directed to generate core meanings, whereas in content dependent learning environment, they are urged to make semantic coding (Barsalou, 1982). Furthermore, Kose et al., (2013) found that using story in teaching programming is also useful. Thus, a child and a UFO meeting story was created in CILE.

CDLE, on the other hand, has been designed following a direct teaching approach. In this environment, abstraction concepts are presented directly using the vocabulary related to abstraction itself. In both environments, learners are provided similar type of links to navigate between the nodes taking the principles of puzzle based approach. Details are provided in the method section.

RESEARCH AIM

This research is two-fold. The first aim of the research was to investigate how students' working memory capacities (WMCs) and different learning environments based on puzzle based learning effect on their abstraction performance. The second one was to investigate whether students' logical reasoning capacities (LRCs) and abstraction ability capacities (AACs) had an effect on students' abstraction performances in each learning environment, when their WMCs were controlled. More specifically, the following research questions were investigated:

1. Are there any main effects of students' WMCs (low and high) and learning environments (CILE and CDLE) on students' abstraction performances?

2. Are there any interaction effects between students' WMCs (low and high) and learning environments (content dependent and content independent) on students' abstraction performances?
3. Are there any effects of students' LRCs and AACs on abstraction performances in CILE and CDLE when WMCs of students are controlled?
4. Do related variables predict students' abstraction performances?

METHOD

RESEARCH DESIGN

180 seventh-grade students in a middle school located in a metropolitan city in Anatolia were invited to the research. In this context, the participants were selected with the convenience sampling method. Fraenkel and Wallen (2012) stated that sample selection is often difficult, and the researcher/s can choose the convenience sampling method in these situations; that the individuals being present for the study.

One of the research questions in this study was to examine how learners' WMCs play a role in understanding the effects of different instructional environments in their achievement. For this reason, the mobile Corsi application was used as a measurement tool to determine participants' WMCs (Uluç & Öktem, 2016). When the obtained data were analyzed, the total score was ranked and the lower and upper groups were determined based on the first and last 27% values. Finally, out of 180, a total of 92 students was invited to the study. These students were randomly assigned to the CDLE and CILE, 46 students in each. The students in the middle group were also attended the sessions, but their data were not used in analyses.

This study is designed as a 2X2 factorial design. Instructional design types (content dependent and content independent) and students' WMCs (low and high) are independent variables whereas abstraction performance is the dependent variable. The students had not received any instruction about abstraction before; therefore, no pretest was performed. Initially the students were separated into two groups, one of which was low working memory group (n=46) and the second was high working memory one (n=46). Students were randomly assigned either to content dependent learning environment (CDLE) or content independent learning environment (CILE) according to their working memory capacities. Four groups were formed; students with low WMCs in CDLE (n=23), students with high WMCs in CDLE (n=23), students with low WMCs in CILE (n=23) and students with high WMCs in CILE (n=23) (See, Table 1). At the end of the instructional intervention, 77 students took abstraction performance test, 15 students did not. For this reason, the study was completed with the participation of 77 students.

Table 1. *The Students Included to Analysis*

WMCs	*Content Dependent Learning Environment*	*Content Independent Learning Environment*
Low	22	18
High	15	22

MATERIALS AND INSTRUMENTS

LEARNING CONTENT

The main objectives of the abstraction were taken from the curriculum proposed by the College Board (AP Computer Science Principles, 2016), where there are three main subject areas (understandings) in the context of abstraction; (1) a variety of abstractions built on binary sequences can be used to represent all digital data, (2) multiple levels of abstraction are used to write programs or create other computational artifacts and (3) models and simulations use abstraction to generate new understanding and knowledge. In this research, the first content area has been addressed, in which the first learning objective is articulated as “to describe the variety of abstractions used to represent data”. Starting from this objective, it is decided to base the content on data representation. Researchers developed six sub learning objectives in accordance with this objective. These are: (1) to describe the importance of abstraction in data representation, (2) to describe how binary systems are used in digital data representation, (3) to perform data representation in binary system, (4) to describe how different number systems are used in the representation of numerical data, (5) to know that the hexadecimal system performs data representation and (6) to convert different number systems to each other.

Three experts in instructional technology were consulted in line with the evaluation of two different learning environments and ask to rate their evaluation over five points for each environment according to the given criteria. Percent agreement was sought among experts over 42 items in the expert opinion form. According to the percentage of congruence analysis (42/42 = 1; concordance = 100%), both learning environments fully met the relevant criteria.

LEARNING APPROACH

Puzzle based learning (PBL) approach was utilized in this study. PBL is an approach mostly based on critical thinking and problem solving (Sooriamurthi et al., 2010). Research with PBL approach showed positive effects on students' motivation and interest (Merrick, 2010; Meyer et al., 2014), allowing critical thinking (Falkner et al., 2012a; Falkner et al., 2012b; Merrick, 2010; Michalewicz et al., 2011), contribution to problem solving skills (Evans & Klymchuk, 2017; Thomas et al., 2013), improving communication and teamwork skills (Forero et al., 2011).

Table 2. *Puzzle Based Learning Approach and Teaching Abstraction*

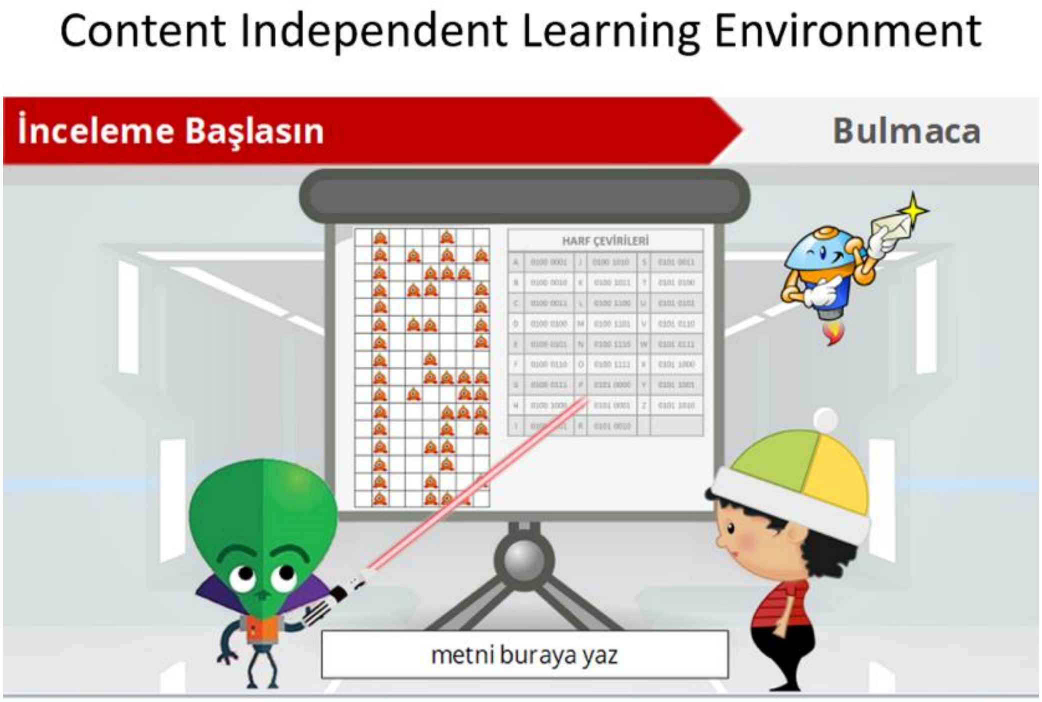
PBL Approach	*Puzzle Based Abstraction Teaching*	*Usage in Learning Environment*
1. Understand the problem 1.1. Take inventory 1.2. Build a model 1.1.3. Draw a diagram	1. Understand the problem 1.1. Take inventory 1.2. Build a model 1.3. Draw a diagram	-A problem situation is given and the student is asked to guess according to the solution principles.
2. Reasoning	2. Let's take a look 2.1. Reasoning 2.2. Pattern recognition	-It is the first part entered in the module. Tips are given about related content.
3. Pattern recognition	3. Let’s go 3.1. Enumerate 3.2. Eliminate 3.3. Simplify	-By presenting the base of the subject, it is expected from student to be able to interpret the content according to the relevant steps.
4. Enumerate and eliminate	4. Let’s think like this	-Activities are presented on how to perform by showing a different dimension related to the
5. Simplify	5. Let’s try like this	topic
6. Perform a gedanken: «what if?» and «so what?»		-The most detailed section of the subject is given here and it is expected that the student will synthesize the previous steps in here.
7. Simulation and optimization 7.1. Simulation 7.2. Optimization	5.1. Simulation 5.2. Optimization	

Meyer et al. (2014) urged instructional designers to apply certain strategies when utilizing PBL approach (understand the problem, reasoning, pattern recognition, enumerate and eliminate, simplify, perform a gedanken, simulation and optimization). These strategies were implemented in a hierarchy (see Table 2), where each step was presented to students under the following headings: understand the problem, let's take a look, let's go, let's think like this, and let's try like this. The environment made it possible for students to proceed to the next topic as they follow the implementation instructions for the puzzles under each topic. These topics and their corresponding functions are presented in Table 2. Expert opinions were obtained for the appropriateness of the approach utilized in the learning environment and interrater reliability analysis was carried out with the percent agreement of two experts' opinions. The percent agreement of experts found to be 79% based on the content criteria presented in Table 2.

LEARNING DESIGN AND LEARNING ENVIRONMENT

Two different learning environments (CILE and CDLE) were designed in Moodle learning management system. These learning environments had the same presentations of preliminary information being given before the puzzles. But the puzzles were differed in being CILE or CDLE. In each learning environment, there were three modules. In every module there were five steps consisted of puzzles which were based on puzzle-based abstraction teaching (see Table 2). Each student couldn't get the next step without solving the puzzles or made mistakes for three times in the same puzzle. Once the puzzle is solved, feedback was provided to the students. If their solution was wrong, students were given two more chances to try to solve the puzzle. In CILE, students are provided the content with a story about a character having a conversation with an alien (see Figure 1). In CDLE, on the other hand, no story was provided (see Figure2). In both contents, students were requested to solve the coded message presented in boxes. The content was presented in three modules, included voiceover with a total of 25-30 screens. Students could navigate forward and backward; and, they could also replay each content from the progress bar on the screen. When solving the puzzles, students were given three trials to complete the task. If they couldn't solve the puzzle, the solution was presented after their third attempt.

Figure 1. *Illustration of a Puzzle for CILE*



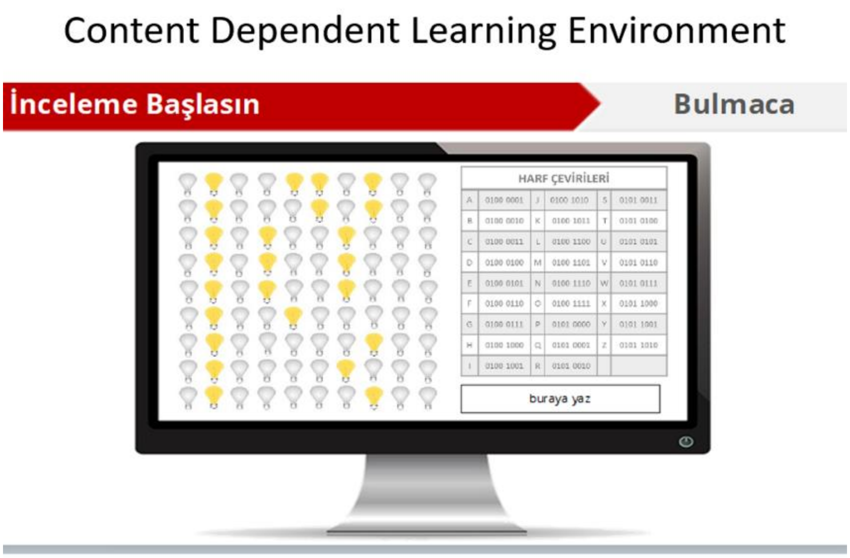
Two experts in the field of instructional technology were consulted to evaluate the puzzles in the learning environments designed according to the PBL approach criteria. Experts evaluated the learning environments with 33 items on a 5-scale evaluation form. Percent agreement analysis (26/33 = 0.79) was conducted to determine the consistency between the scores of the experts. Accordingly, the fit value of the two raters was calculated as 79%.

DATA COLLECTION

MEASURING WMCS

Corsi Block Tapping test is a psychological test that measuring the one's working memory (Brunetti et al., 2014). The backward-recall and forward-recall tasks of the test have a relation with the working memory (Berch et al., 1998; Vandierendonck et al., 2004). In this manner, a mobile Corsi Block Tapping application, adapted for tablet PC presentation on a 12.2 inch screen, was used to measure students' WMCs. This mobile Corsi Block Tapping is one of the iapplications from the test battery of Computer Applied Intelligence Screening, which was developed for the children been aged between 6-16, adapted and validated by Uluç and Öktem (2016). In this Corsi application, there are two tasks called Corsi flat and Corsi reverse with a trial and practice sessions. In each task, nine blue squares as a block are placed at their relative standard positions on a white background. Each block in turn is highlighted by changing its colour to green for 1 s. with an inter block time of 500 ms.

Figure 2. *Illustration of a Puzzle for CDLE*



In Corsi flat, a standard trial starts when the participant touch the trial tab. Then an announcement is provided to start the trial. Next, the sequence of blocks is highlighted and the presentation ends with a warning sound. Immediately after this sound, the participant can repeat the sequence by touching the squares sequentially in any order. An announcement comes after this repetition to warn the participant whether the sequence is correct or not. If the participant wants to start the practice, touches the start tab on the screen. After an announcement, practice begins with two selected block sequences. Each sequence requires the participant to repeat this sequence15 times. In practice session, no feedback is provided. This process continues with increasing one block more at each sequence. If the participant makes mistakes two times consecutively, his/her process is terminated. In Corsi reverse, the process is the same as Corsi flat. But inversely, it is expected form the participant to reverse the sequence. For example, presentation begins with two blocks and when it's the participant's turn, s/he is expected to touch the boxes with reverse order.

Once the tasks were completed, data are saved as .txt (plain text) file for each participant, with true and false responses. The total numbers of right responses from both tasks were calculated and the results constituted the working memory scores.

MEASURING LOGICAL REASONING COMPETENCIES

In order to measure students' LRCs, a logical reasoning test (LRT) was applied to 169 7th grade students. The test included 12 items during the development stage. An item analysis procedure was executed and one item was excluded because of its discrimination value being less than 0.20. Cronbach Alpha value of the test was found .68 with the reliability analysis. The values obtained from the item analysis are presented in Table 3. During the development of the test, expert opinions were taken to determine the suitability of the test for 7th grade students. In this direction, two field experts (from the dep. of Computer and Instructional Technologies) presented their opinions in the context of the subject content, and two field experts (from the dep. of Psychology) in the context of skills.

MEASURING ABSTRACTION ABILITY

In order to examine students' abstraction skills, a task similar to the cypher task in the WISC-R intelligence scale, was formed in the form of a paper and pencil test. It was aimed to determine the abstraction skill levels of students with this task, which had consisted of three sub-tasks. An expert opinion was sought from a faculty at the psychology department. This task required students to match certain symbols corresponding to provided symbols. Cronbach Alpha value of the test was found .98 with the reliability analysis. 138 secondary school students have participated in this test. Two lecturers, from psychology department, guided for developing this test to provide the suitability for 7th grade students.

Table 3. *Item Difficulty and Discrimination of the Logical Reasoning Test*

	Q01	*Q02*	*Q03*	*Q04*	*Q05*	*Q06*	*Q07*	*Q08*	*Q09*	*Q10*	*Q11*
Item Difficulty	0,73	0,41	0,53	0,56	0,56	0,39	0,66	0,46	0,41	0,43	0,50
Item Distinctiveness	0,33	0,56	0,48	0,58	0,63	0,56	0,60	0,38	0,65	0,65	0,63

Table 4. *Item Difficulty and Discrimination of the Cypher Task*

Item	*Item Difficulty*	*Item Distinctiveness*	*Item*	*Item Difficulty*	*Item Distinctiveness*	*Item*	*Item Difficulty*	*Item Distinctiveness*
Q01	0,81	0,38	Q12	0,69	0,62	Q23	0,68	0,65
Q02	0,81	0,38	Q13	0,72	0,57	Q24	0,70	0,59
Q03	0,81	0,38	Q14	0,68	0,65	Q25	0,74	0,51
Q04	0,77	0,46	Q15	0,73	0,54	Q26	0,72	0,57
Q05	0,74	0,51	Q16	0,69	0,62	Q27	0,65	0,70
Q06	0,70	0,59	Q17	0,72	0,57	Q28	0,64	0,73
Q07	0,76	0,49	Q18	0,72	0,57	Q29	0,68	0,65
Q08	0,76	0,49	Q19	0,70	0,59	Q30	0,66	0,68
Q09	0,72	0,57	Q20	0,68	0,65	Q31	0,68	0,65
Q10	0,73	0,54	Q21	0,72	0,57	Q32	0,64	0,73
Q11	0,73	0,54	Q22	0,66	0,68			

Short informative session was held with each student to show how to proceed in the test. Once they completed successfully, they are permitted to start the test. The time between the start and the end is kept in seconds. 32 symbols in the respective task are presented to students and they were asked to match each symbol by drawing the corresponding symbols in sequence. The same rules apply for the students who take the second task after the first performance task is completed. In the third task, students are asked to establish the relationship between the symbols presented in the first and second performance tasks and to perform matching accordingly. While performing this task, the time in seconds is also recorded. During the evaluation process, only the correct numbers of the last task were taken into account. During the development of the cypher task, item analyses and reliability analysis were carried out for 32 symbols to which matches were made. The values obtained from item analysis are presented in Table 4.

MEASURING ABSTRACTION PERFORMANCE

An abstraction performance test with 13 items was developed to determine how many learning outcomes students achieved. Two field experts, from the dep. of Computer and Instructional Technologies, presented their opinions in the context of the subject content for being suitable to 7th grade students. Before the actual application of the test, a pilot study group was formed with students who had studied abstraction before. An item analysis was conducted to the data (see Table 5). The analyses revealed that only item Q2 was between 0.20 and 0.30, so this item was rephrased and then the test was finalized. The Cronbach Alpha value was found to be .73. This value is seen to be as acceptable (Taber, 2017). This test was utilized once the teaching practice was completed.

PROCEDURE

ASSESSMENT OF WMCS

A total of 169 middle school students took Mobile Corsi working memory tool. Once the data collected, it was sorted from low to high. Lower and higher groups were determined based on upper and lower 27% scores. Finally, a total of 92 students were separated into two groups representing low and high WMCs groups. The students from upper and lower groups were randomly assigned to four different learning environments according to their working memory capacities. Thus, there were 23 students in each of the four groups. Students' name, last name and school number information were requested from the school administration; then, user accounts were created for two learning environments on Moodle. Ethic committee approval was presented to the school administration and the related partners.

LEARNING SESSION

The pilot study was conducted with 343 middle school students. Teaching process was lasted six weeks. Students took orientation for the learning environment in the first week. Researchers were not physically present during the process. However, they had constant contact with teachers during the process. The students were provided to access learning environments in computer laboratories using external headphones with the guidance of their teachers. At the same time, students also entered the learning environment in their free time. Teachers told researchers about the difficulties they had faced. Thus, that was allowed the environment to be much better. Pilot abstraction performance test was carried out in the sixth and the last week of the study. 316 students took the abstraction performance test.

The main study was conducted with 92 seventh grade middle school students. Teaching process was lasted six weeks. Students took orientation for the learning environment in the first week. Then, four weeks of learning process started. Students accessed to learning environments in the computer labs on the relevant course day and hour. Each student interacted with one computer and had an ear plug. Students, who were outside the study group but in the same classes with the study group, also entered the program and passed the same process. The teachers guiding the process got in touch with the researchers frequently in every step and tried to prevent possible problems. Abstraction performance test was carried out in the sixth week. 77 students participated in the abstraction performance test.

FINDINGS

STUDENTS' DESCRIPTIVE DATA

A cross tabulation analysis was formed representing gender variables and the relationship between other variables including the chi square values. The ownership of technological tools by students showed no statistically significant gender differences for mobile phones, smartphones, tablets, netbooks, laptop computers and desktop computers according to gender. When it comes to the using purposes of technological tools according to gender, no statistically significant gender differences found for in terms of search, communication, and play. However, it was seen that using the technological tools for playing purposes had statistically significant positive but weak relationship gender difference in favor of boys (see Table 5).

LEARNING ENVIRONMENTS AND STUDENTS' ABSTRACTION PERFORMANCE

Abstraction performance test results were analyzed by ANOVA. Prior to ANOVA, assumptions were checked and ensured no violations existed. Descriptive statistics and ANOVA results are presented in Table 6 and 7 respectively. As it is seen in Table 6 and Table 7 that students' WM levels have main effects on their abstraction performance ($F(1, 73)= 9.299$, $p<.05$) in favor of high WM group ($X=7.22$, $SS=2.32$) on the students' abstraction performance scores (see Table 7) whereas no statistically significant effect of learning environment was found ($F(1, 73)= 0.786$, $p>.05$). In other words, regardless of being exposed in CILE or CDLE, students' abstraction performance did not change. Finally, no interaction effect of learning environments and WMCs on the students' abstraction performance scores was observed ($F(1, 73)= 0.418$, $p>.05$).

WORKING MEMORY CAPACITIES AND STUDENTS' ABSTRACTION PERFORMANCE

Different WMCs are effective on learning, and students with high WMCs are significantly more successful than students with low WMCs (See Table 6 and Table 7). The findings also showed that CILE and CDLE had no main effects on learning, WMCs and interaction effects were not significant. In order to explore whether further variables intervene the performances, follow up analyses were carried out. In the following sections, these analyses will be presented along with the discussions in the field of individualized learning differences.

Table 5. *Ownerships and Usage Reasons*

Items	*Positive responses* Girl (n=37)	*Boy (n=40)*	*Pearson Chi Square* Value	*Statis. Sig.*	*Cramer's V* Value	*Statis. Sig.*
Mobile (basic) phone ownership	8	16	3,026	,082		
Smartphone ownership	34	38	,306	,580		
Tablet ownership	28	28	,312	,576		
Netbook ownership	4	5	,053	,818		
Laptop computer ownership	21	28	1,457	,227		
Desktop computer ownership	12	16	,476	,490		
Using for searching	33	34	,298	,585		
Using for communication	33	31	1,872	,171		
Using for entering social media	34	30	3,908	,048		
Using for playing	26	37	6,385	,012	,288	,012

Table 6. *Descriptive Statistics*

	WMCs High			*Low*			*Total*		
Environment	X	SD	N	X	SD	N	X	SD	N
Content Independent	6,86	2,23	22	5,50	2,60	18	6,25	2,47	40
Content Dependent	7,33	2,43	15	5,64	2,57	22	6,49	2,69	37
Total	7,22	2,32	37	5,58	2,55	40	6,36	2,56	77

STUDENTS' AACS AND LRCS AND ABSTRACTION PERFORMANCES

Initial results showed that CDLE and CILE had no effects on students' abstraction performances. As a follow up analysis related to the third Research question, an ANCOVA analysis was performed to explore whether students' LRCs and AACs had any effect on their abstraction performance within each learning environment. Since WM had yielded a main effect, MW scores had been taken as a covariate in the analysis. The skewness and kurtosis values of each of the groups to be compared were examined to see if the scores of the dependent variable showed normal distribution, and the ratio of these values to their standard errors was within the limits of ± 1.96; it was determined that Kolmogorov Smirnov was normally distributed according to the significance values (p> .05). Levene's test was performed to determine the equality of variances and it was seen that they were distributed homogeneously (p> .05). To find out whether there was a linear relationship between the dependent variable and the control variable, simple linear regression was run and it was determined that there was a linear relationship. In order to determine the homogeneity of the regression coefficients in the groups, customized model was performed in the covariance analysis. Here, the regression coefficients of the groups were found to be homogeneous (p> .05). ANCOVA was carried out to determine whether the control and independent variables were independent from each other.

Table 7. *Results of Variance Analysis*

Source	*df*	*Sum of Squares*	*F*	*p*	*Ŋ2*	*Power Size*
Abstraction performance						
Environment	1	4,749	,786	,378	,011	,141
Working memory	1	56,190	9,299	,003	,113	,853
Environment * Working memory	1	2,523	,418	,520	,006	,098
Error	73	6,043				

Table 8. *Students' LRCs on Their Abstraction Performances in CDLE*

GROUP	*N*	*Mean*	*Adjusted Mean*
Low LRCs	10	4,700	4,842
Middle LRCs	16	6,375	6,669
High LRCs	11	8,273	7,716

Table 9. *Effects of students' LRCs on Their Abstraction Performances in CDLE: ANCOVA Results*

Source	*Sum of Squares*	*df*	*Mean of Squares*	*F*	*Sig.*
Working Memory	11,213	1	11,213	2,024	,164
LRCs	39,206	2	19,603	3,538	,041
Error	182,818	33	5,540		
Total	261,243	*36*			

The adjusted values of the means of the abstraction performance scores in CDLE across students' LRCs were presented in Table 8. ANCOVA results (See Table 9) yielded that there was a significant effect of LRCs on students' abstraction performance [F(1-36)= 19.603, p<.05]. According to the Bonferroni test result, it was observed that students with high LRCs (X=7.716) had a significant difference against students with low LRCs (X=4.842). There was no significant difference found between students with middle LRCs (X=6.669) and the other groups.

Table 10. *Students' AACs on Their Abstraction Performances in CDLE*

GROUP	*N*	*Mean*	*Adjusted Mean*
Low AACs	8	6,500	6,563
Middle AACs	14	*6,*143	6,391
High AACs	15	*6,*800	*6,*535

Students' AACs were also investigated in CDLE. Descriptive statistics regarding the adjusted means of abstraction performance scores across their AAC levels were presented in Table 10. ANCOVA results yielded no significant effect of AACs on their abstraction performance [F(1-36)= .102, p>.05] (See Table 11).

Table 11. *Effects of students' AACs on Their Abstraction Performances in CDLE: ANCOVA Results*

Source	*Sum of Squares*	*df*	*Mean of Squares*	*F*	*Sig.*
Working Memory	36,295	1	36,295	42,968	,026
AACs	,205	2	,102	5,400	,985
Error	221,820	33	6,722	,015	
Total	261,243	36			

The adjusted values of the means of the abstraction performance scores in CILE across students' LRCs were presented in Table 12. ANCOVA results yielded that there was a significant effect of AACs on students' abstraction performance [F(1-39)= 24.103, p<.05] (See Table 13). According to the Bonferroni test result, it was observed that students with high LRCs (X=7.528) had a significant difference against students with low LRCs (X=4.783). There was no significant difference found between students with middle LRCs (X=6.499 and the other groups. Students' AACs were also investigated in CILE. Descriptive statistics regarding the adjusted means of abstraction performance scores across their AAC levels were presented in Table 14. ANCOVA results yielded no significant effect on their abstraction performance [F(1-39)= 4.545, p>.05] (See Table 15).

Table 12. *Students' LRCs on Their Abstraction Performances in CILE*

GROUP	*N*	*Mean*	*Adjusted Mean*
Low LRCs	13	4,769	4,783
Middle LRCs	15	6,400	6,499
High LRCs	12	7,667	7,528

Table 13. *Effects of students' LRCs on Their Abstraction Performances in CILE: ANCOVA Results*

Source	*Sum of Squares*	*df*	*Mean of Squares*	*F*	*Sig.*
Working Memory	13,689	1	13,689	2,884	,098
LRCs	48,206	2	24,103	5,078	,011
Error	170,885	36	4,747		
Total	237,500	39			

Table 14. *Students' AACs on Their Abstraction Performances in CILE*

GROUP	*N*	*Mean*	*Adjusted Mean*
Low AACs	13	6,846	6,861
Middle AACs	17	5,647	5,750
High AACs	10	6,500	6,306

INVESTIGATING PREDICTOR VARIABLES ON STUDENTS' ABSTRACTION PERFORMANCES

Regression analysis was performed to reveal if any predictor variables and a prediction model could be provoked to understand students' abstraction performance. exist. WMCs and LRCs, which had a significant effect on students' abstraction performance, were included into regression analysis. Multiple linear regression was carried out to determine the extent to which LRCs and WMCs had an impact on students' abstraction performance. Assumption tests for multiple regression were run and examined prior to performing the analyzes. The ratios of skewness and kurtosis values of LRCs and WMCs were within ± 1.96 limits. It has been observed that the predictor variables have a linear relationship with the predicted variable. The predicted variables were independent from each other and the difference between the estimated values and the observed values was determined to be normally distributed. According to the analysis results presented in Table 16, there was a moderate relationship between LRCs and WMCs (R=0.460, R2=0.211, F(2-74)=9.924, p<0.01). These two variables explained 21% of students' abstraction performance scores.

Table 15. *Effects of students' AACs on Their Abstraction Performances in CILE: ANCOVA Results*

Source	Sum of Squares	df	Mean of Squares	F	Sig.
Working Memory	16,075	1	16,075	2,756	,106
AACs	9,091	2	4,545	,779	,466
Error	210,00	36	5,833		
Total	237,500	39			

According to the standardized regression coefficients, the relative importance order of the predictive variables on students' abstraction performance were in order as LRCs (β = .381) and WMCs (β =.187). Considering the significance tests of the regression coefficients, only LRCs (p <.01) among the predictive variables showed that students' abstraction performances were significantly predictive. It was observed that the WMCs (p> .05) of the students did not significantly predict their abstraction performance. When the relationship between LRCs and students' abstraction performance was investigated, (r = .422, other predictive variables were controlled, r = .386) correlation was observed. Regression equation predicting students' abstraction performance is modeled as follows:

abstraction performance = 3.056 + (.364 * LRCs)

Table 16. *Related Variables Predicting Students' Abstraction Performance*

Source	*B*	*Std. Error*	*β*	*T*	*Sig.*	*R*	*Partial R*
Constant	3,056	,818	-	3,736	,000	-	-
Logical Reasoning	,364	,101	,381	3,602	,001	,422	,386
Working Memory Capacity	,084	,047	,187	1,771	,081	,271	,202
R=,460 R2=,211							
F(2-74)=9,924 p=,000							

DISCUSSION, CONCLUSION AND IMPLICATIONS

Cognitive individual differences in education are important. It was found that WMC has statistically significant main effect on abstraction performance in favor of students with high WMCs. The students with low WMCs, on the other hand, should be provided instructional support to compensate their learning performance. When these differences are not considered, no effect of instructional guidance is observed, which makes it confusing to discuss PBL or its strategies whether it would make a difference. Furthermore, more research is needed to observe whether any other cognitive individual differences exist on abstraction performance.

In this research it was first observed that learning environments had no statistically significant main effects on students' abstraction performance scores. Secondly, when the main effects of the WMCs on students' abstraction performance were examined, a statistically significant effect was observed in favor of the students with high WMCs. Furthermore, when students' WMCs were controlled, it was seen that students' LRCs had a significant effect on their abstraction performances for CILE and CDLE. Then abstraction performances were compared and it was found that students with high LRCs had a significant difference against students' with low LRCs for both environment. But there was no significant difference found between middle and low LRC groups. Although there are various definitions in the literature that emphasize the importance of reasoning skill in the context of abstraction, research investigating the role of reasoning in abstraction teaching is limited (see Duncan, 2013; Saitta & Zucker, 2001; Zucker, 2003). The LRCs positively predicted students' abstraction performance; yet, it was found that students' mere WMCs did not predict their abstraction performance. For further studies, test of LR is needed to be studied to increase the level of reliability.

The learning environments presented within the scope of the abstraction teaching had no significant effect on the abstraction performance alone. Furthermore, it was observed that students with high WMCs and LRCs had significantly higher effects on abstraction performance. This showed that students with low skills did not compensate their abstraction performance within existing learning environments. In this context, it can be said that the CILE and CDLE did not significantly contribute on the abstraction performances of students with different cognitive skills. In addition, it had revealed that logical reasoning is a variable that predicts abstraction performance in abstraction teaching. However, it was observed that the WMCs alone was not a predictor of students' abstraction performance. So this revealed the assumption that learning environments can provide advantages to students with high WMCs. Also it is uncertain whether the abstraction performances of the students with low WMCs in the CILE and CDLE presented in the study were lower than the abstraction performances of the students with high WMCs due to the teaching strategy (PBL approach) used.

In this research it was found that students' abstraction ability had no effect on students' abstraction performances for both environments. As there are studies that reported no relationship between abstraction skills and learning (Bennedsen & Caspersen, 2006), it was concluded that different group lessons created were affected by abstraction skills compared to standard courses Bennedsen & Caspersen, 2008). So, it can be stated that there is a need for further studies to investigate abstraction abilities and abstraction in CS.

Regardless of the provided learning environments, the results of this study showed that students' abstraction performances were modulated by their WMCs. Thus, this result indicates that students with low WMCs could not overcome their limitations when left alone in a learning environment. Using only puzzle-based learning approach in this research could cause no statistically significant differences via learning environments, where no main or interaction effects with WMCs were observed. When the literature about abstraction teaching is examined in the context of learning environments; it was observed that only Bennedsen and Caspersen (2008) carried out a research that investigating groupings which were content based, structure of observable learning outcome (SOLO) level and atomic grouping. They found that grouping variables (content based, structured-solo and structured-atomic) had significant effects on students' programming performances. To sum up, when individual differences are taken into account, instructional designers could consider other variables in addition to working memory capacity when choosing an instructional method. Instructional method alone fails to capture every student, especially when there are certain cognitive differences exist.

When literature is examined, abstraction was mainly studied with university students as participants (see Bennedsen & Caspersen, 2008; Hazzan & Kramer, 2007; Hill et al., 2008). Studies with high school students (Sakhnini & Hazzan, 2008; Taub et al., 2014), middle school students (Statter & Armoni, 2017; Statter & Armoni, 2020), and elementary school students (Çakıroğlu et al., 2021; Statter & Armoni, 2017) are limited. Thus, further research is needed for each participant group so that the results obtained for teaching abstraction can be generalized and meta-analyzes can also be done. In this manner, the research of Tikva and Tambouris (2021) was also important to show this need. Their research aim was to develop a conceptual model, for computational thinking in programming in K12 education, by a systematic literature review. The results showed that in literature review researchers mention about abstraction but it can not been seen that the studies which focuses on abstraction are missing. Mirolo (2022) had some suggestion on studying abstraction with different student groups.

The first was computational thinking for primary school students, the second was developing awareness of the role of multiple abstraction levels and the last one was working on complex systems when learning programming.

Bennedsen and Caspersen (2008) claimed that the definitions of abstraction in CS textbooks and how they are applied are explained in general statements and not elaborated. Also, Zucker (2003) pointed out that there were various problems in the context of understanding the abstraction, among which is the definition of abstraction. Existing and varying definitions could also lead to increased complexity (see Duncan, 2013; Hazzan & Tomayko, 2005; Hill et al., 2008); hence, it makes it more difficult for teachers, as well. Therefore, there is also a need to specify a unified definition of abstraction for learners and teachers in CS curricula.

Abstraction can be taught as early as possible in CS courses (Bucci et al., 2001). Having taken WMCs and its effect on students' abstraction performances, educators should pay more attention to designing instructional tasks when teaching abstraction especially at middle grades. Although more research is needed to explore the developmental trajectories of WM when CS curriculum is implemented at middle grades, it is important to emphasize that students' WMCs are so important that it cannot be ignored. More research is needed to explore the reciprocal relationship in a CS learning environment. In this manner, Sentance and Csizmadia (2017) produced five key themes, that put forth from 300 CS teachers' opinions, respectively are unplugged activities, contextualization of tasks, collaboration, developing computational thinking and scaffolding programming. Further research can be designed based on these constructs.

Since WM has a developmental trajectory, Kramer's (2007) recommendation is well taken when urging to measure students' abstraction skills annually. Since abstraction is an important component of CS curriculum (Bennedsen & Caspersen, 2008; Haberman & Muller, 2008; Hazzan and Tomayko, 2005), it should be measured at different abstraction forms, at different abstraction levels and for different purposes of abstractions. Therefore, more tools are needed to measure abstraction. Similarly, measuring WM requires special training and takes time to employ it on an individual base, more practical tools and/or implicit measurement methods need to be developed.

This study showed that students' WMCs have main effect on their abstraction performances, but the designed learning environments have not. Abstraction uses so many common complex cognitive tasks such as language, comprehension, learning, reading, problem solving, reasoning and planning (see Baddeley, 1983; Baddeley, 1992; Cowan, 2014). Thus, much more studies are needed to investigate the relationship between both and other related variables.

In this study, one of the learning environments has been designed based on puzzle-based learning environment design principles. Since no interaction effect between the learning environment and WMCs was observed, researchers could explore the interaction effect with other environments, such as project based and problem based learning environments.

ACKNOWLEDGEMENT

This research has been a part of the doctoral thesis conducted by the first author under the supervision of the second author.

AUTHOR CONTRIBUTION

- The first author worked on the design, implementation and reporting of the research and also writing the manuscript.
- The second author provided guidance and evaluation in the context of field expertise at all stages of the process. Also revised the manuscript in the context of language, order and flow.

REFERENCES

AP Computer Science Principles (2016). *Course and exam description: effective fall 2016.* CollegeBoard: New York. [Retrieved from: https://secure-media.collegeboard.org/digitalServices/pdf/ap/ap-computer-science-principles-course-and-exam-description.pdf, Date of access: October 9th, 2016].

Armoni, M. (2013). On teaching abstraction in cs to novices. *Journal of Computers in Mathematics and Science Teaching, 32*(3), 265-284.

Altun, A. (2012). Ontologies for personalization: a new challenge for instructional designers. *Procedia-Social and Behavioral Sciences, 64,* 691-698.

Baddeley, A. (1983). Philosophical transactions of the royal society of London. *Series B, Biological Sciences, 302*(1110), Functional Aspects of Human Memory (Aug. 11, 1983), 311-324.

Baddeley, A. (1992). Working memory. *Science, 255*(5044), 556-559.

Barsalou, L. W. (1982). Context-independent and context-dependent information in concepts. *Memory & Cognition, 10*(1), 82-93.

Bennedsen, J., & Caspersen, M. E. (2006). Abstraction ability as an indicator of success for learning object-oriented programming?. *ACM SIGCSE Bulletin, 38*(2), 39-43.

Bennedsen, J., & Caspersen, M. E. (2008). Abstraction ability as an indicator of success for learning computing science. *In Proceedings of the Fourth International Workshop on Computing Education Research,* 15-26. ACM.

Berch, D. B., Krikorian, R., & Huha, E. M. (1998). The Corsi block-tapping task: Methodological and theoretical considerations. *Brain and Cognition, 38*(3), 317-338.

Bergersen, G. R., & Gustafsson, J.-E. (2011). Programming skill, knowledge, and working memory among professional software developers from an investment theory perspective. *Journal of Individual Differences, 32*(4), 201–209.

Brunetti, R., Del Gatto, C., & Delogu, F. (2014). eCorsi: implementation and testing of the Corsi block-tapping task for digital tablets. *Frontiers in Psychology, 5*, 939.

Bucci, P., Long, T. J., & Weide, B. W. (2001). Do we really teach abstraction?. *In ACM SIGCSE Bulletin, 33*(1), 26-30. ACM.

Caspersen, M. E., & Bennedsen, J. (2007). Instructional design of a programming course: a learning theoretic approach. *In Proceedings of the Third International Workshop on Computing Education Research,* 111-122. ACM.

Colburn, T., & Shute, G. (2007). Abstraction in computer science. *Minds and Machines, 17*(2), 169-184.

Costa, J. M., & Miranda, G. L. (2019). Using alice software with 4C-ID Model: effects in programming knowledge and logical reasoning. *Informatics in Education, 18*(1), 1-15.

Cowan, N. (2014). Working memory underpins cognitive development, learning, and education. *Educational Psychology Review, 26*(2), 197-223.

CSTA. (2011). Computational thinking in K–12 education teacher resources second edition [Retrieved from: https://csta.acm.org/Curriculum/sub/CurrFiles/472.11CTTeacherResources_2ed-SP-vF.pdf, Date of access: October 9th, 2016].

Çakıroğlu, Ü., Çevik, İ., Köşeli, E., & Aydın, M. (2021). Understanding students' abstractions in block-based programming environments: A performance based evaluation. *Thinking Skills and Creativity, 41,* 100888.

Çetin, İ., & Dubinsky, E. (2017). Reflective abstraction in computational thinking. *The Journal of Mathematical Behavior, 47,* 70-80.

Duncan, J. (2013). The structure of cognition: attentional episodes in mind and brain. *Neuron, 80*(1), 35-50.

Evans, T., & Klymchuk, S. (2017). Enhancing generic problem-solving and thinking skills of tertiary stem students: Through puzzle-based learning. spaces and pedagogies. *New Zealand Tertiary Learning and Teaching Conference 2017 Proceedings.*

Falkner, N., Sooriamurthi, R., & Michalewicz, Z. (2012a). Teaching puzzle-based learning: development of basic concepts. *Teaching Mathematics and Computer Science, 10*(1), 183-204.

Falkner, N., Sooriamurthi, R., & Michalewicz, Z. (2012b). Teaching puzzle-based learning: development of transferable skills. *Teaching Mathematics and Computer Science, 10*(2), 245-268.

Forero, A., & et al. (2011). Complexity in design of digital systems: active learning with puzzles. *Proceedings of the 7th International CDIO Conference.* 2011.

Fraenkel, J. R., & Wallen, N. E. (2012). *How to design and evaluate research in education* (8th edition). Mc Grawall Hill.

Guarino, L. R. (1978). *The evolution of abstraction in programming languages* (No. CMU-CS-78-120). Carnegie-Mellon Univ Pittsburgh Pa Dept Of Computer Science.

Haberman, Br., & Muller, O. (2008). Teaching abstraction to novices: Pattern-based and ADT-based problem-solving processes. *In 2008 38th Annual Frontiers in Education Conference,* F1C-7 – FIC-12. IEEE.

Hazzan, O. (2003). How students attempt to reduce abstraction in the learning of mathematics and in the learning of computer science. *Computer Science Education, 13*(2), 95-122.

Hazzan, O., & Kramer, J. (2007). Abstraction in computer science & software engineering: a pedagogical perspective. *Frontier Journal, 4*(1), 6-14.

Hazzan, O., & Tomayko, J. E. (2005). Reflection and abstraction in learning software engineering's human aspects. *Computer, 38*(6), 39-45.

Hill, J., & et. al. (2008). Applying abstraction to master complexity. *In Proceedings of the 2nd International Workshop on the Role of Abstraction in Software Engineering,* 15-21. ACM.

Kawash, J. (2012). Engaging students by intertwining puzzle-based and problem-based learning. *In Proceedings of the 13th Annual Conference on Information Technology Education,* 227-232.

Kemp, H. (2014). Exploring a story-based learning design in a grade 4 science and technology classroom *(Dissertation, Nelson Mandela Metropolitan University).*

Kose, U., Koc, D., & Yucesoy, S. A. (2013). Design and development of a sample" computer programming" course tool via story-based e-learning approach. Educational Sciences: Theory and Practice, 13(2), 1235-1250.

Kramer, J. (2007). Is abstraction the key to computing?. *Communications of the ACM, 50*(4), 36-42.

Kuloğlu, N., & Asasoğlu, A. O. (2010). Indirect expression as an approach to improving creativity in design education. *Procedia-Social and Behavioral Sciences, 9,* 1674-1686.

Melham, T. (2013). Modelling, abstraction, and computation in systems biology: a view from computer science. *Progress in Biophysics and Molecular Biology, 111*(2-3), 129-136.

Merrick, K. E. (2010). An empirical evaluation of puzzle-based learning as an interest approach for teaching introductory computer science. *IEEE Transactions on Education, 53*(4), 677-680.

Meyer III, E., & et. al. (2014). *Guide to Teaching Puzzle-Based Learning.* Springer.

Michalewicz, Z., Falkner, N., & Sooriamurthi, R. (2011). Puzzle-based learning: an introduction to critical thinking and problem solving. *Decision Line, 42*(5), 6-9.

Mirolo, C., Izu, C., Lonati, V., & Scapin, E. (2022). Abstraction in Computer Science Education: An Overview. *Informatics in Education, 20*(4), 615-639.

Oberauer, K. (2009). Design for a working memory. *Psychology of Learning and Motivation, 51,* 45-100.

Oyelere, S. S., & et. al. (2019). Impact of puzzle-based learning technique for programming education in Nigeria context. *In 2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)* (2161), 239-241. IEEE.

Passey, D. (2017). Computer science (CS) in the compulsory education curriculum: Implications for future research. *Education and Information Technologies, 22*(2), (2017): 421-443.

Rich, K. M., Yadav, A., & Zhu, M. (2019). Levels of abstraction in students' mathematics strategies: what can applying computer science ideas about abstraction bring to elementary mathematics?. *Journal of Computers in Mathematics and Science Teaching, 38*(3), 267-298.

Saitta, L., & Zucker, J. D. (2001). A model of abstraction in visual perception. *Applied Artificial Intelligence, 15*(8), 761-776.

Sakhnini, V., & Hazzan, O. (2008). Reducing abstraction in high school computer science education: the case of definition, implementation, and use of abstract data types. *Journal on Educational Resources in Computing (JERIC), 8*(2).

Salleh, T. S. A., & Zakaria, E. (2011). Integrating computer algebra systems (CAS) into integral calculus teaching and learning at the university. *International Journal of Academic Research, 3*(3), 397-401.

Saribatur, Z. G., Eiter, T., & Schüller, P. (2021). Abstraction for non-ground answer set programs. *Artificial Intelligence, 300*, 103563.

Sentance, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. *Education and Information Technologies, 22*(2), 469-495.

Sigel, I. E. (1953). Developmental trends in the abstraction ability of children. *Child Development,* 131-144.

Sooriamurthi, R., Falkner, N., & Michalewicz, Z. (2010). Puzzle-based learning. *Journal of Computing Sciences in Colleges, 25*(3).

Sooriamurthi, R., Falkner, N., & Michalewicz, Z. (2012). Teaching abstraction in mathematics and computer science - a computer-supported approach with alloy. *CSEDU (2),* 239-245.

Sprague, P., & Schahczenski, C. (2002). Abstraction the key to CS1. *Journal of Computing Sciences in Colleges, 17*(3) (2002): 211-218.

Statter, D., & Armoni, M. (2017). Learning abstraction in computer science: a gender perspective. *In Proceedings of the 12th Workshop on Primary and Secondary Computing Education,* 5-14. ACM.

Statter, D., & Armoni, M. (2020). Teaching abstraction in computer science to 7th grade students. *ACM Transactions on Computing Education (TOCE), 20*(1), 1-37.

Taber, K. S. (2018). The use of Cronbach's alpha when developing and reporting research instruments in science education. *Research in Science Education, 48*(6), 1273-1296.

Taub, R., Armoni, M., & Ben-Ari, M. M. M. (2014). Abstraction as a bridging concept between computer science and physics. *In Proceedings of the 9th Workshop in Primary and Secondary Computing Education*, 16-19. ACM.

Thomas, C., & et. al. (2013). Puzzle-based learning of mathematics in stem subjects. *The Higher Education Academy.*

Tikva, C., & Tambouris, E. (2021). Mapping computational thinking through programming in K-12 education: A conceptual model based on a systematic literature review. *Computers & Education, 162*, 104083.

Uluç, S., & Öktem, F. (2016). TÜBİTAK SOBAG 112K428 "Bilgisayar Uygulamalı Zeka Tarama Testi" Basılmamış Proje Sonuç Raporu (15/04/2013 - 07/01/2016).

Vandierendonck, A., Kemps, E., Fastame, M. C., & Szmalec, A. (2004). Working memory components of the Corsi blocks task. *British Journal of Psychology, 95*(1), 57-79.

Waite, J. L., & et. al. (2018). Abstraction in action: K-5 teachers' uses of levels of abstraction, particularly the design level, in teaching programming. *International Journal of Computer Science Education In Schools.*

Wolz, U., & Conjura, E. (1994). Abstraction to implementation: a two stage introduction to computer science. *Proceedings of the Annual National Educational Computing Conference.* 1994.

Zehetmeier, D., & et. al. (2019). Defining the competence of abstract thinking and evaluating CS-students' level of abstraction. *In Proceedings of the 52nd Hawaii International Conference on System Sciences.*

Zucker, J. D. (2003). A grounded theory of abstraction in artificial intelligence. *Philosophical Transactions of the Royal Society of London B: Biological Sciences, 358*(1435), 1293-1309.