Computers in Human Behavior 114 (2021) 106575

Contents lists available at ScienceDirect

Computers in Human Behavior

journal homepage: http://www.elsevier.com/locate/comphumbeh

Full length article

An adaptive educational computer game: Effects on students' knowledge and learning attitude in computational thinking

Danial Hooshyar [a], Liina Malva [a], Yeongwook Yang [a,*], Margus Pedaste [a], Minhong Wang [b,c], Heuiseok Lim [d]

[a] Institute of Education, University of Tartu, Estonia
[b] Faculty of Education, The University of Hong Kong, Hong Kong
[c] Department of Educational Information Technology, East China Normal University, Shanghai, China
[d] Department of Computer Science and Engineering, Korea University, Seoul, Republic of Korea

ARTICLE INFO

Keywords:
Adaptive educational computer game
Computational thinking knowledge and skills
Adaptive learning
Adaptive game-play

ABSTRACT

Several studies have reported that adaptivity and personalization in educational computer games facilitate reaching their full educational potential. However, there is little effort to develop adaptive educational computer games for promoting students' computational thinking (CT). In this study, an adaptive computer game is introduced, called AutoThinking, that not only promotes both CT skills and conceptual knowledge, but also provides adaptivity in both game-play and learning processes. To evaluate the possible effects of the game, an experimental study was carried out with 79 students in an elementary school in Estonia. AutoThinking and a conventional technology-enhanced learning approach were used for teaching CT to the experimental and control group, respectively. Our results reveal that AutoThinking improved students' CT skills and conceptual knowledge better than the conventional approach. It was also found that students with a low and high level of prior knowledge made higher improvement in knowledge gain using the adaptive game compared to the traditional approach, especially those students with lower prior knowledge. Finally, our findings show that the adaptive game could also improve students' learning attitude toward CT better than the conventional approach, especially those students with higher prior learning attitudes.

1. Introduction

One of the main skills that future generations must develop is Computational Thinking (CT). CT, a cutting-edge term defined by Wing (2006), is regarded as a cognitive ability that enables people to develop computational solutions for a current problem by applying computer sciences' reasoning processes. Application of CT has gone even beyond STEM (science, technology, engineering, and mathematics) domains, and according to several research, CT plays crucial roles in other disciplines as well, for example, medicine, digital humanities, computational finance, archaeology, economics, and so forth (National Research Council, 2010; Selby & Woollard, 2013; Wing, 2014). Multiple studies have reported that CT's integration into curricula would benefit both cognitive and non-cognitive aspects of learning (e.g., Brown, Sentance, Crick, & Humphreys, 2014, pp. 1–22; Haddad & Kalaani, 2015; Malva, Hooshyar, Yang, & Pedaste, 2020; Repenning et al., 2015, p. 11; Román-González, Pérez-González, & Jiménez-Fernández, 2017). Therefore, several countries all over the world have made reformation of educational programs on different educational levels in order to integrate CT into their official curricula (e.g., Brown et al., 2014, pp. 1–22; Carlborg, Tyren, Heath, & Eriksson, 2019; DR, 2018; Perković, Settle, Hwang, & Jones, 2010; Wing, 2011).

These rapid changes have resulted in some challenges, however. For instance, to keep up teachers with experience and knowledge within the area, to develop suitable methods paired with learning materials, and to properly manage limited resources to educate teachers in CT (Angeli & Giannakos, 2020; Brown et al., 2014, pp. 1–22; Färnqvist, Heintz, Lambrix, Mannila, & Wang, 2016). On the other hand, there are some other challenges in fostering CT in educational practice, including lack of motivation and opportunities to promote students' CT. For example, research conducted by Yardi and Bruckman (2007) reports that students often show negative attitudes toward learning CT, which could result in

* Corresponding author.
E-mail addresses: danial.hooshyar@gmail.com (D. Hooshyar), liina.malva@ut.ee (L. Malva), yeongwook.yang@gmail.com (Y. Yang), margus.pedaste@ut.ee (M. Pedaste), magwang@hku.hk (M. Wang), limhseok@korea.ac.kr (H. Lim).

https://doi.org/10.1016/j.chb.2020.106575
Received 29 March 2020; Received in revised form 17 September 2020; Accepted 18 September 2020
Available online 20 September 2020
0747-5632/

impeding its proper development. Therefore, different approaches have to be developed and applied to not only make CT more available and engaging to learners, but also assist teachers in promoting students' CT. One prospective solution could be educational computer games.

Educational games, or game-based learning approaches, have gained researchers' attention as they have proven to be effective learning tools that engage and motivate students, and also improve their learning achievements (e.g., Hooshyar, Yousefi, Wang, & Lim, 2018; Pontes, Duarte, & Pinheiro, 2020; Zumbach, Rammerstorfer, & Deibl, 2020). Even though there exist several educational games aimed at promoting students' CT, to a large extent, they ignore fostering CT skills. Instead, they mostly support reinforcing CT's conceptual knowledge and students' motivation (e.g., Hooshyar et al., 2019; Kazimoglu, Kiernan, Bacon, & Mackinnon, 2012a; Kuruvada, Asamoah, Dalal, & Kak, 2010; Zhao & Shute, 2019). On the other hand, even more importantly, the existing CT games ignore providing adaptivity in the game. Despite several calls from researchers and practitioners to focus more on personalization and adaptation that would suit the individual needs of the player, the existing CT games mainly follow predefined and rigid computer-assisted instruction concepts. This could impede exploring their full educational potential (e.g., Hooshyar, Yousefi, & Lim, 2018; Kickmeier-Rust, Mattheiss, Steiner, & Albert, 2011; Ku, Hou, & Chen, 2016; Peirce, Conlan, & Wade, 2008).

Considering the importance and relevance of CT in society together with the existing gap in CT game research, an adaptive educational game was developed, called AutoThinking, for teaching CT (both learning CT conceptual knowledge and skills), while engaging students with individually tailored game-play (Hooshyar et al., 2019). To evaluate the effectiveness of the adaptive game compared to a conventional technology-enhanced learning approach (i.e., a lesson using a PowerPoint presentation with Multimedia delivered by a teacher), an experimental study on elementary school students was conducted. In line with this aim, our research questions were designed as the following:

(1) Does the adaptive game improve students' skills and conceptual knowledge in CT better than the traditional technology-enhanced learning approach?
(2) What is the effect of different learning approaches on CT knowledge gain of the students with different prior knowledge?
(3) What is the effect of different learning approaches on the learning attitude of the students with different prior learning attitudes?

The outline of this paper is as follows: Section 2 and 3 review the related work and introduce the adaptive game, respectively. Section 4 describes the methodology, while Section 5 revolves around the results. Section 6 provides the discussion and conclusions.

2. Related research

Although the motive behind the movement of computational thinking (CT) can be traced back to several researchers during the past half century—e.g., DiSessa, 2000; Papert, 1980, 1996; Wilensky, 2001—Wing (2006) for the first time employed the term "computational thinking". According to Wing's definition (2014, para. 5), CT is "the thought processes involved in formulating a problem and expressing its solution(s) in such a way that a computer—human or machine—can effectively carry out". In other words, CT refers to characterization of the thinking process of computer scientists while trying to solve a problem. Wing (2006) argues that CT includes a set of competencies that alike numeracy and literacy should be learned and acquired by every person in early education.

Despite the long history of CT research, no solid consensus among researchers has been reached about the definition of CT and what processes or competencies it comprises (e.g., Angeli & Giannakos, 2020; Angeli & Valanides, 2020; Guzdial, 2008). More information on overview of the emergence of CT definitions and skills has been given by Palts and Pedaste (2020). For example, Wing (2006) has defined six main dimensions for CT, namely problem formulation, abstraction, problem reformulation, problem decomposition, automation, and systematic testing. Furthermore, Denning (2009) argues that CT goes beyond computer science and is about seven different views or categories, namely coordination, communication, computing, recollection, design and assessment, and automation. Ater-Kranov, Bryant, Orr, Wallace, and Zhang (2010) have explored the importance of different computation thinking skills and concluded that the application of abstractions to solve a problem, along with algorithmic thinking are the key competencies of CT. Ater-Kranov et al. (2010), unlike Wing (2006), found that since complex CT can also occur spontaneously, engineering and mathematical thinking are not essential parts of CT. On the other hand, according to a National Research Council report on CT's scope, multiple skills encompassing CT occur, including problem decomposition, parallelism, debugging, search strategy, and simulation (National Research Council, 2010, p. 3). In another study, Brennan and Resnick (2012) report that CT includes several concepts and skills (practices) of which results in the development of CT knowledge. According to their argument, CT skills include abstracting and modularizing, pattern recognition, debugging, and simulation, whereas CT concepts include sequences, loops, parallelism, events, conditionals, operators, and data. Barr and Stephenson (2011) state that CT consists of several skills, among them problem decomposition, data collection, data analysis, data representation, simulation, automation, algorithms and procedures, and parallelism. Although there exists a lack of consensus among researchers on the different components of CT, there are four different skills and three concepts that several studies agree on. The skills consist of problem solving or algorithmic thinking (including identifying and decomposing the problem), building algorithms (including creating efficient and repeatable patterns), debugging, and simulation, while the concepts are sequence, conditional logic, and loop logic. These skills and concepts might have been defined differently by researchers, however, their importance as core elements of CT is becoming ubiquitously accepted one way or another (Brennan & Resnick, 2012; El Mawas, Hooshyar, & Yang, 2020; Kazimoglu, Kiernan, Bacon, & MacKinnon, 2012b; Tsarava et al., 2017; Wing, 2006; Zhao & Shute, 2019).

Another challenge, besides clarifying the definition and elements of CT, that is worthy of mentioning is the misconception that programming (or coding) and CT are identical. Even though multiple studies state that programming is not a synonym for CT (e.g., Barr, Harrison, & Conery, 2011; Guzdial, 2008; Repenning, n.d.; National Research Council, 2010; Repenning, Webb, & Ioannidou, 2010; Wing, 2006), yet many instructors think that these two are equal (e.g., Blum & Cortina, 2007). In this regard, there exist several studies using learning environments or games developed for early programming as a tool to teach CT (e.g., Wu's Castle (Eagle & Barnes, 2009), CodeCombat (Saines, Erickson, & Winter 2013), MiniColon (Ayman, Sharaf, Ahmed, & Abdennadher, 2018)). According to many researchers, among them Hooshyar et al. (2019), Grover and Pea (2013), and Zhao and Shute (2019), alignment of environments that are developed to teach programming (coding in particular) is incomplete with CT. More specifically, alignment of such tools might not be able to meet the need for fostering learners' CT competence. One reason is that while using these environments, learners get distracted and overwhelmed by syntax of the programming languages presented to them in different forms, such as blocks. Instead of using coding to teach CT, such environments should focus more on solving a problem through meaningful conceptualization and activation of thought processes (Yadav, Mayfield, Zhou, Hambrusch, & Korb, 2014, pp. 1–16). One way to do so is to offer mapping between problems and their alternative solutions. More importantly, tools aimed at fostering CT should provide learners with appropriate and context familiar feedback during development of a solution for a given problem. We consequently should differentiate between tools that teach programming and coding, and those that systematically support fostering learners' CT competence.

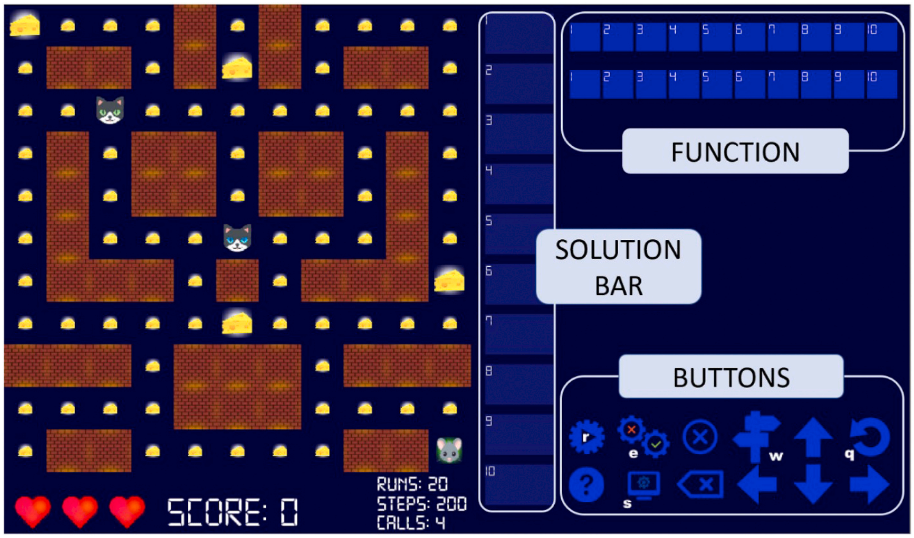

Fig. 1. Interface of AutoThinking.

One prospective method to promote CT is educational computer games which have proven to be helpful learning tools that not only have the potential to engage and motivate students, but also to improve their learning achievements (e.g., Hwang, Chiu, & Chen, 2015; Sung & Hwang, 2013; Vandercruysse et al., 2016; Vos et al., 2011). In the literature, there are several learning environments that use coding in order to teach CT to students, as computer programming and CT share overlapping aspects in terms of cognitive skills. For example, Scratch (Resnick et al., 2009), Snap! (Harvey & Mönig, 2010), and Blockly (Fraser, 2013) mostly use block-based and visual programming, or they adapt game design principles in order to reduce the complexity associated with programming languages syntax. To do so, they mostly adapt drag-and-drop interactions in order to offer simpler solutions than writing syntax. Although such environments have shown success in boosting learners' motivation and learning achievement in early programming, they suffer from some issues. On one hand, these environments, that are often called games for promoting CT, cannot be reckoned as educational games as they lack several relevant elements of educational games, such as timely feedback, supporting engagement, enhancing retention, and incentives (e.g., Kazimoglu, Kiernan, Bacon, & MacKinnon, 2013; Kickmeier-Rust et al., 2011). On the other hand, according to several studies, e.g., Brennan and Resnick (2012) and Meerbaum-Salant, Armoni, and Ben-Ari (2011), they are incapable of promoting a deeper level of learning. One reason is that the alignment of these environments with CT skills is partial and incomplete, and learners are not provided with enough opportunities to involve in conceptualization and basic taught processes of solving problems.

There are some other games that, by benefiting from different elements of educational games, support learning different skills, among them CT (e.g., Weintrop & Wilensky, 2012). Basically, educational games aimed at promoting CT use motivating contexts to engage players in the process of developing solutions to a problem. These educational games are capable of supporting more meaningful learning due to different game elements, in contrast to block-based and visual programming environments (e.g., Land, 2000). For example, Eagle and Barnes (2009), Esper, Foster, Griswold, Herrera, and Snyder (2014), and Ayman et al. (2018) developed Wu's Castle, CodeSpell, and MiniColon to teach early programming or supporting knowledge of CT, respectively. There exist several studies reporting the positive impact of such tools on learners' programming skills and CT. Similar to most of the other tools, however, these environments use programming languages that are text-based which begs the substantial attention of learners to syntax details (Zhao & Shute, 2019). Therefore, they are also not fully aligned with CT. Besides, some of these environments fail to provide learners with enough opportunities to develop CT skills as their main focus is promotion of abstract and conceptual knowledge of CT (Hooshyar et al., 2019; Kazimoglu et al., 2012a). To promote CT skills, Kazimoglu, Kiernan, Bacon, and Mackinnon (2012) developed a game and evaluation of their game showed that it can improve learners' motivation, and could also be beneficial in improving learners' CT skills. Similarly, Zhao and Shute (2019) developed a video game for fostering students' CT, especially their CT skills. Evaluation of their game revealed that playing the game for even less than 2 h significantly improved the students' CT skills, whereas no influence was found on their attitudes toward CT. Moreover, there are some other educational games that target promoting CT, for instance RoboBuilder (Weintrop & Wilensky, 2012), LightBot (Gouws, Bradshaw, & Wentworth, 2013), and games at Code.org. Studies reporting the effects of these games on learners are preliminary or qualitative with small sample sizes, requiring further investigations and experimentations (e.g., Giannakoulas & Xinogalos, 2018; Weintrop & Wilensky, 2012).

Besides existing gaps highlighted so far, the educational games aimed at fostering CT mainly tend to neglect adaptation and personalization to individual needs. In other words, such games follow unadaptable and rigid computer-assisted instruction concepts, which could impede exploring their full educational potential (e.g., Kickmeier-Rust et al., 2011; Peirce et al., 2008; Xie, Chu, Hwang, & Wang, 2019). Even though the application of educational games to CT have shown to be effective, there is still some room for improvements in such games.

3. AutoThinking

AutoThinking[1] is an adaptive educational game developed for promoting students' skills and conceptual knowledge in CT (Hooshyar et al., 2019). AutoThinking, to the best of our knowledge, is the first adaptive educational game developed for fostering students' CT that includes adaptivity in both game-play and learning process. With the aim of excluding syntactical errors that in return reduces the learners' cognitive load, AutoThinking uses icons rather than syntax of computer programming languages (see Fig. 1).

In AutoThinking players need to develop different types of strategies and solutions to complete three different levels (see Table 1). As the player is in the role of a mouse, he/she needs to collect all cheese pieces

[1] http://www.autothinking.ut.ee/.

Table 1
Description of the three different levels of AutoThinking.

Level	Big cheese (total no.)	Moving big cheese (no.)	Cats (no.)	Cats eye color	Adaptivity	Game features									CT skills targeted	CT concepts targeted
						Solution bar	Arrow buttons	Run button	Help button	Conditional button	Loop button	Simulation button	Function bar	Debug button		
Level 1	2	0	1	green	no	✓	✓	✓	✓						- Problem identification and decomposition	-Sequence
Level 2	2	0	1	Blue	no	✓	✓	✓	✓	✓	✓	✓			- Problem identification and decomposition - Simulation	- Sequence - Conditional - Loop
Level 3	4	2	2	green and blue	Yes	✓	✓	✓	✓	✓	✓	✓	✓	✓	- Problem identification and decomposition - Building algorithms - Debugging - Simulation	- Sequence - Conditional - Loop

and score as much as possible, and at the same time escape from two cats in the maze. Players can develop up to 20 solutions for clearing all 76 cheese pieces in the maze. During the game-play, the player receives more points for solutions that involve various CT concepts or skills, and travelling non-empty tiles. Note that AutoThinking offers different options to players so as to develop different solutions for the current state of the maze. For instance, the game is featured with a "function" bar allowing players to save various patterns and solutions, and if necessary apply or generalize in different situations of the game (see Fig. 1). Before running their solution, players should thoughtfully consider the movements of the cats as they are prone to create risky situations. More explicitly, one cat moves intelligently as many tiles as traveled by the mouse, whereas the other cat moves randomly with iteration through the maze the same amount of tiles as the number of commands used in the "solution bar" by the player. The game provides players with various types of feedback (textual, graphical, or video) and hints, if necessary, based on the suitability of the solution for the current state of the maze.

In a unique way, AutoThinking promotes four CT skills, and three CT concepts (for more details, see Hooshyar et al., 2019). The connection between activities in the game and different skills and concepts of CT are listed in Table 2.

3.1. Adaptivity in AutoThinking

After the player executes a solution, the game data is inputted to the Bayesian Network (BN) algorithm which has been developed by experts in the field (for more details, see Hooshyar et al., 2019). Accordingly, the BN decides which algorithm the cat should follow for the current solution of the player and, if necessary, what kind of feedback or hint should be provided to the player.

3.1.1. Adaptivity in game-play

According to the quality of the developed solution by the player, during game-play, one of the cats moves intelligently, while the other one still behaves randomly. For the intelligent cat, the game takes into account three different condition at the same time: whether the solution is risky in terms of the mouse getting caught by cats, whether it has the potential to gain a good score, and whether player use proper CT skills or concepts in their solution for current state of the maze. Accordingly, the probabilistic-based decision-making technique used in the game, BN, automatically assesses players' skills and controls the cat movements in four different ways: 1) regulates the cat to move randomly without iteration through the maze, 2) regulates the cat to move provocatively by going close to the mouse (up to one tile away), not to catch it, and come back, 3) regulates the cat to move aggressively aimed at catching the mouse (by finding the shortest distance from the mouse), and 4) regulates the cat not to get closer than six tiles away from the mouse. To switch between the four algorithms, the cat takes into account both the short and long term solutions of the player (considering both the current and previous solution developed by the player). The other cat, however, still behaves randomly with repetition based on the number of commands placed in the solution bar. This makes AutoThinking an interesting and unpredictable game that always provides the player with a new situation that might not have occurred to previous players.

3.1.2. Adaptivity in learning

The game provides players with timely textual, graphical, or video feedback about CT concepts and skills, that are embedded in the game-play, based on the current state of the maze and the player's skill level (long- and short-term). To enable players to improve their solutions, the game also highlights some of the game features or buttons as hints (see Fig. 2).

Adaptivity in learning is provided by the game in two different phases, before or after running the solution. The player can directly "run" the solution without using the debug button, resulting in timely adaptive feedback or hints, after running the game, which would help

Table 2
Game activities and features related to CT skills and concepts.

	Category	Game activity	Game feature
CT SKILLS	Problem identification and decomposition (algorithmic thinking)	Helping the mouse to collect all cheese pieces and score as much as possible, and at the same time escape from cats in the maze.	The "solution bar", "function bar", and different buttons in Table 2 are designed to let players develop solutions, using sequence of proper actions, and generalizable patterns where they can be used in different situations of the game.
	Building algorithms (creating efficient and repeatable patterns)	Using "functions bar" different rules/strategies can be generalized or re-used multiple times.	The "solution bar", "function bar", and different buttons in Table 2 are designed to let players develop solutions, using sequence of proper actions, and generalizable patterns where they can be used in different situations of the game.
	Debugging	Using "debug" button the current solution algorithm can be debugged/monitored.	The "debug" button give players chance to monitor the solution and possibly identify any errors in its logic.
	Simulation	Using "simulation" button players can practice run-time mode.	The "simulation" button let players simulate their solution before actually running it to observe the outcome of their solution regardless of intervention of other variables in the game, such as the cats' movements and cheese pieces.
CT CONCEPTS	Sequence	Using different buttons (arrows, loop, conditional) players can create sequence of proper actions and see the execution after executing.	The "solution bar", "function bar", and different buttons in Table 2 are designed to let players develop solutions, using sequence of proper actions, and generalizable patterns where they can be used in different situations of the game.
	Conditional	Using "conditional" button players can decide based on specific conditions, i.e., If there are junctions, stop at junction N, else continue going.	The "conditional" button let players decide based on certain conditions that support the result of multiple outcomes.
	Loop	Using "loop" buttons players can keep repeating the same sequence N times, practicing loop logic.	The "loop" button let players run the same sequence of actions multiple times.

the player to know about the shortcomings and mistakes in previous solutions and possible ways to overcome them. Such adaptivity which aims to promote several CT skills and concepts individually supports learners in developing the most optimum solution for the problem in hand. On the other hand, players can utilize the "debug" button to similarly activate the decision-making technique and receive the estimation of the suitability of their solution. This way players are given a chance to, if necessary, change and improve their solution to have a more optimum solution.

4. Research design

4.1. Participants and experimental process

Participants of this experimentation were elementary school students in Estonia, aged 11 and 12 years old (fifth grade). Two classes of students (16 girls and 20 boys) were randomly assigned to the experimental group where adaptive game was used as the learning approach, while the other two classes (20 girls and 23 boys) were assigned to the control group using a traditional technology-enhanced learning approach (i.e., a lesson using a PowerPoint presentation with Multimedia delivered by a teacher) to learn CT. A typical class size of an elementary school in Estonia usually varies from 15 to 25 students, which was also the case of this study. In order to prevent possible influence of different teachers on the outcome of this study, all lessons were conducted by the same teacher. Students were informed that participation in this experimentation is not mandatory and would have no influence on their final grades. Ethical approval from the ethical committee of (removed for blind review) (issue date: November 18, 2019; Application Registration Number: 298/T-7) was obtained to carry out this study. All students and their parents have signed consent forms to participate in our study before starting the experiment.

Our experiment started with 30–45 min of answering the pre-test for measuring CT knowledge (including both conceptual knowledge and skills) and a pre-questionnaire measuring students' attitude towards learning CT. Afterward, both groups of students participated in 60–75 min of learning sessions using different learning approaches. The control group learned CT skills and conceptual knowledge from a teacher using a PowerPoint presentation with Multimedia, whereas the experimental group used AutoThinking for learning the CT skills and concepts. For both groups, the CT skills and concepts covered during the learning activity were identical. More explicitly, the students learned about sequence, loop, and conditional concepts, as well as algorithmic thinking, pattern recognition, debugging, and simulation skills. For the control group, CT content was presented by a teacher and supported with class discussions and multimedia examples. Experimental group, on the other hand, used all three levels of the AutoThinking game for learning CT content, while the teacher had an assistive role (e.g., providing technical help and basic instructions of the game). Lastly, once the learning activity was finished, the post-test and learning attitude questionnaire were answered by students (30–45 min).

4.2. Instrument and data collection

Instruments used in this study include a pre- and post-test, and pre- and post-questionnaire of learning attitude. The main aim of the pre- and post-test was to examine students' CT knowledge (including concepts and skills) before and after the experiment. These tests included nine multiple-choice items and one fill-in-the-blank item. The pre- and post-test items were adapted from a validated instrument for assessing CT developed by González (2015), and were augmented and examined by two experienced teachers and researchers with over 5 years of experience in teaching and researching CT and computer science courses. In order to find out which skills and concepts have the highest influence on the results, three groups of items for both skills and concepts were created (see Tables 3 and 4).

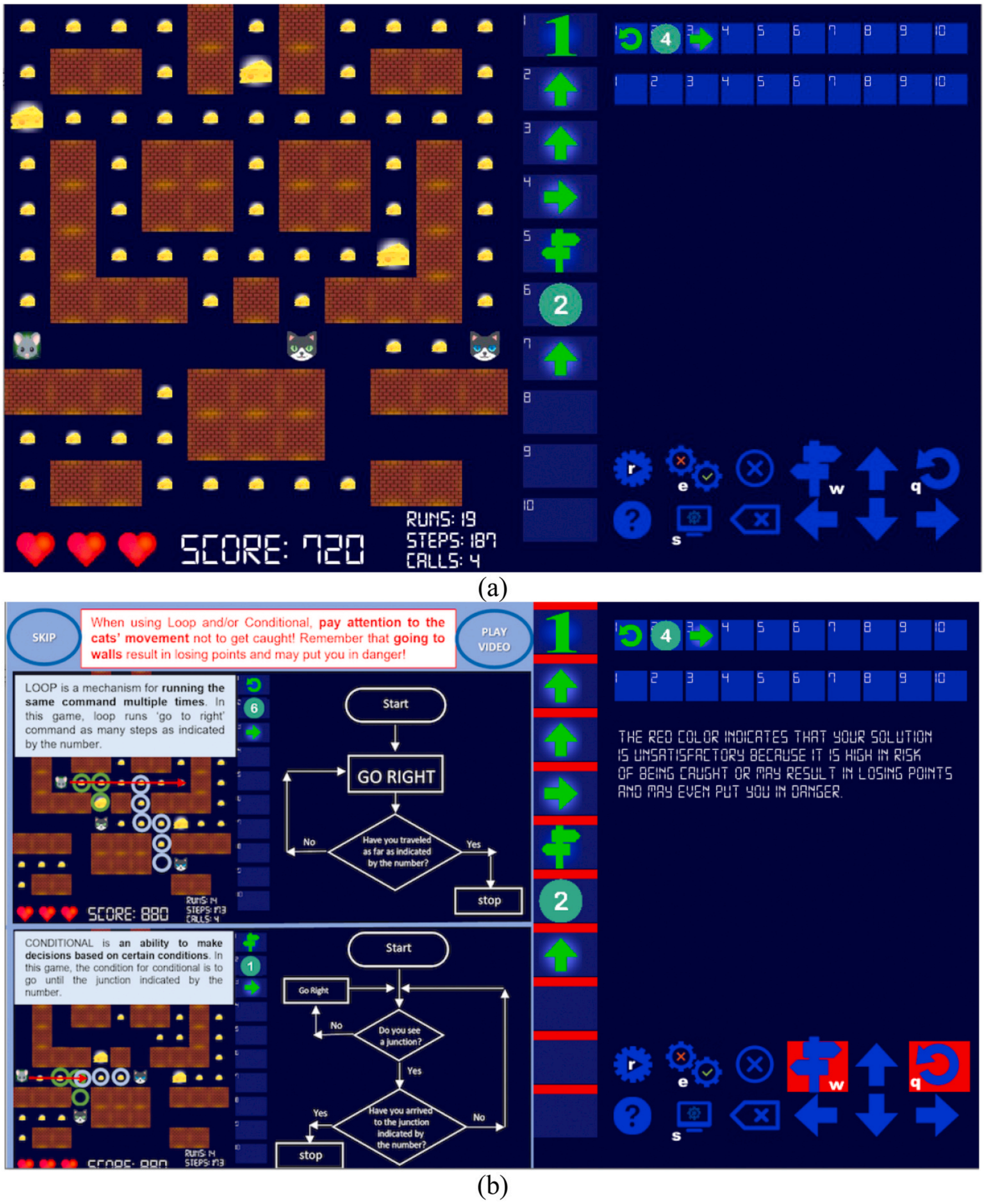

Fig. 2. (a) A solution developed by a player, and (b) debugging of the developed solution (feedback and hints).

Table 3
Skills targeted in three categories of items.

Skills group	Number of items	CT skills targeted			
		Algorithmic thinking	Pattern recognition	Debugging	Simulation
4-skill	3	✓	✓	✓	✓
3-skill	3	✓	✓		✓
2-skill	4	✓			✓

Table 4
Concepts targeted in three categories of items.

Concepts group	Number of items	CT concepts targeted		
		Sequence	Conditional logic	Loop logic
3-concept	2	✓	✓	✓
2-concept	5	✓		✓
1-concept	3	✓		

As shown in Table 3, there were three items targeting all four skills simultaneously, three items targeting three skills at the same time, and four items targeting two skills. As seen from the table, algorithmic thinking (including identifying and decomposing the problem) and simulation are the skills that are part of every single item. In Fig. 3a, an item is presented where all the four CT skills are simultaneously targeted (problem identification and decomposition, pattern recognition, simulation, and debugging).

With regard to the CT concepts, shown in Table 4, there were two items targeting all three concepts simultaneously, two items targeting two concepts at the same time, and three items targeting only one concept. The sequence concept is covered in all of the items. In Fig. 3b, an item is presented where all the three CT concepts are simultaneously targeted (sequence, conditional logic, and loop logic).

A validated questionnaire of learning attitude, adapted from Hwang and Chang (2011), was used for both pre- and post-questionnaires. The questionnaire included five items that had to be answered on a 5-point Likert scale (1 – fully disagree; 5 – fully agree). "It is worth learning things about computational thinking" and "I would like to learn more

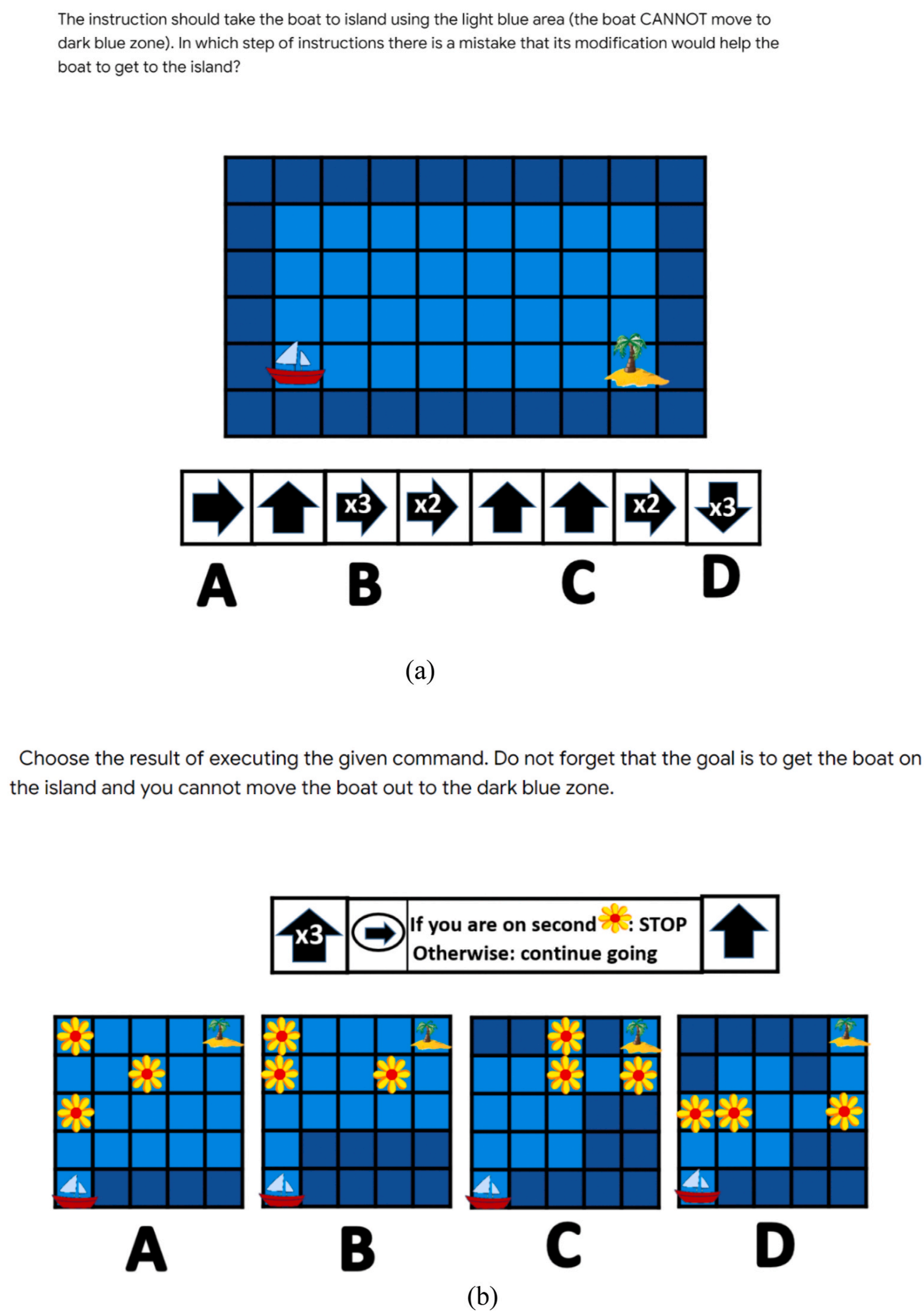

Fig. 3. Two examples from pre- and post-test: (a) an item where all four CT skills are targeted, and (b) an item where all three CT concepts are targeted.

Table 5
Statistical description of different groups of students.

Groups	N	Mean	SD
Experimental	36	8.94	1.19
Control	42	8.14	1.35

Table 6
ANCOVA analysis of overall CT knowledge.

Source	Sum of squares	Df	Mean square	F	Sig.	Partial η^2
Group	7.50	1	7.50	6.20	.01*	.07
Error	90.70	75	1.20			

*$p < .05$.

about computational thinking" are two examples of items in this questionnaire. The value of Cronbach's alpha for the questionnaire of learning attitude is 0.77 and 0.81 for pre- and post-questionnaires, respectively.

5. Results

5.1. Overall CT knowledge gain

To investigate the effectiveness of the AutoThinking game with regard to enhancing students' overall CT knowledge, the pre- and post-test scores were used (all ten questions in the pre- and post-test). Due to missing data, one student from the control group was excluded. Moreover, as suggested by Field (2009), a few existing outliers were identified and replaced by the next-highest score plus one unit. Data from 78 students was analyzed using the one-way ANCOVA with the pre-test scores, post-test scores, and the learning approach as covariate, dependent, and independent variable, respectively. As there exist slight differences between the pre-test scores of both groups in terms of mean and standard deviation (experimental group: $M = 8.11$, $SD = 1.59$; control group: $M = 7.67$, $SD = 1.16$) and such differences may still affect the post-test performance, one-way ANCOVA was adopted. This would help to obtain more precise statistical results to explain the CT knowledge gain of students, as the pre-test scores were excluded. To do so, first the basic assumptions of ANCOVA were checked. Accordingly, the regression coefficient indicated no significant interaction between the independent variables and the covariate (F (1,74) = 3.30, p = .07). Additionally, the result of Levene's test was not found to be significant (F (1,76) = 0.50, p = .48), indicating that between the groups there exists a residual variance homogeneity, and therefore this study can employ the ANCOVA test to identify differences among groups. Furthermore, a Pearson correlation test was run on the pre-test and post-test for the overall CT knowledge gain. The result shows that there is a significant positive correlation between covariate and the dependent variable (r = 0.53; p = .000). According to the ANCOVA results, shown in Tables 5 and 6, both groups' post-test scores were significantly different when the effect of their pre-test scores were excluded, F (1,75) = 6.20, p = .01, with a small effect size (Cohen, 1988). The mean and adjusted mean scores of the control group were 8.14 and 8.22 (SE = 0.17), respectively, whereas for the experimental group, the mean and adjusted mean scores were 8.94 and 8.85 (SE = 0.18), respectively. This finding shows the effectiveness of the AutoThinking game compared to the traditional technology-enhanced learning approach in improving students' overall CT knowledge.

5.2. CT skill gain

To explore the possible effect of the AutoThinking game with regard to enhancing students' CT skills, items were divided into three groups that targeted all four skills, three skills or two skills (see section instrument and data collection). To this end, data from 78 students was analyzed (36 in the experimental and 42 in the control group) using multivariate analysis of covariance (MANCOVA) with the learning approach as independent variable, the 4-skill, 3-skill, and 2-skill items in post-test scores as dependent variables and pre-test scores as covariate, respectively. The MANCOVA approach was selected to address alpha error inflation. Before conducting the test, first the assumptions of MANCOVA were checked, including homogeneity of regression and homogeneity of variance-covariance matrices. For different types of learning approach, the tests for homogeneity in the regression coefficients of the covariate were found to be non-significant, indicating the appropriateness of the common regression coefficient for the covariance portion of the analysis. The assumption of homogeneity of variance was met for the 3-skill (F (1,76) = 0.48, p = .49) and 2-skill items (F (1,76) = 1.81, p = .18), whereas for the 4-skill this assumption was violated (F (1,76) = 11.59, p = .01). To correct the violation of homogeneity, as recommended by Tabachnick, Fidell, and Ullman (2007), a more stringent significance level was used in the analysis (i.e. 0.01). Furthermore, the assumption of homogeneity of variance-covariance was found to be significant (Box's M = 21.68, P < .05), therefore, Pillai's Trace was selected for interpretation of the results (Allen & Bennett, 2008). Finally, with regards to the correlation between variables, as reported previously (see section 5.1), the covariates and dependent variables were positively correlated at significance level of 0.01.

According to the MANCOVA result, there were significant differences between the experimental and control groups on the combined post-test CT skill variables, after controlling for pre-tests: F (3, 72) = 4.74, p = .00, Pillai's Trace = 0.17, partial $\eta^2 = 0.17$. The MANCOVA results for each dependent variable are illustrated in Table 7. A separate examination of the results reveals that there are significant differences in scores between both groups with regards to the 4-skill items (F (1, 74) = 6.78, p = .01, partial $\eta^2 = 0.08$) and 3-skill items (F (1, 74) = 7.53, p = .00, partial $\eta^2 = 0.09$). Nonetheless, for the 2-skill items, not significant differences were found between groups, F (1, 74) = 1.41, p = .23, partial $\eta^2 = 0.02$. For the combined dependent variables, an effect size of 0.17 was found. That is, 17% of the variance in the dependent variables was explained by the independent variables. Furthermore, effect size for the 4-skill, 3-skill, and 2-skill items were 0.08, 0.09, and 0.02, respectively. The effect size of the combined dependent variables and the variables separately were small. In 4-skill items, both the mean and adjusted mean score of the experimental group were 2.86 (SE = 0.08), and the mean and adjusted mean score of the control group were 2.57 (SE = 0.07). Furthermore, the mean and adjusted mean score for the experimental group in 3-skill items were 2.50 and 2.48 (SE = 0.10), and for the control group they were 2.09 and 2.10 (SE = 0.09), whereas for the experimental group in 2-skill items the mean and adjusted mean were 3.77 and 3.76 (SE = 0.09), and for the control group in 2-skill items they were 3.59 and 3.60 (SE = 0.08). Even though in 2-skill items there were no significant differences between the groups, the adjusted means of both groups reveal that students in the experimental group show a higher improvement than the control group. Regarding the 4- and 3-skill items, the experimental group significantly outperformed the control group in acquiring CT skills. Overall, these findings suggest the effectiveness of the AutoThinking game compared to the traditional technology-enhanced learning approach in improving students' CT skills.

5.3. CT conceptual knowledge gain

To explore the possible effect of the AutoThinking game with regard

Table 7
MANCOVA analysis of CT skills.

Skill (items no)	Control group		Experimental group		F	P	Partial η^2
	Mean	SD	Mean	SD			
4-skill (3)	2.57	.63	2.86	.35	6.77	.01*	.08
3-skill (3)	2.10	.69	2.48	.50	7.52	.00*	.09
2-skill (4)	3.60	.62	3.76	.54	1.41	.23	.01

*p < .01.

Table 8
MANCOVA analysis of CT conceptual knowledge.

Concept (items no)	Control group		Experimental group		F	P	Partial η^2
	Mean	SD	Mean	SD			
3-concept (2)	1.57	.50	1.83	.37	6.91	.01*	.08
2-concept (5)	3.90	.98	4.27	.84	3.17	.07	.04
1-concept (3)	2.73	.44	2.83	.37	.91	.34	.01

*p < .01.

Table 9
Statistical description of students' prior knowledge.

Learning approach	Groups	N	Mean	SD
Experimental	Lower prior knowledge	12	8.41	1.08
	Higher prior knowledge	24	9.33	.81
Control	Lower prior knowledge	17	7.23	1.03
	Higher prior knowledge	25	8.80	1.08

to enhancing students' CT conceptual knowledge, from the ten pre- and post-test items, items were divided into three groups that targeted all three concepts, two concepts or one concept. To this end, data from 78 students were analyzed using MANCOVA with the learning approach as independent variable, the 3-concept, 2-concept, and 1-concept items in post-test scores as dependents and pre-test scores as covariate, respectively. Before conducting the test, first the assumptions of MANCOVA were checked, including homogeneity of regression and homogeneity of variance-covariance matrices. For different types of learning approach, the tests for homogeneity in the regression coefficients of the covariate were found to be non-significant, indicating the appropriateness of the common regression coefficient for the covariance portion of the analysis. The assumption of homogeneity of variance was met for the 2-concept (F (1,76) = 1.15, p = .29) and 1-concept items (F (1,76) = 3.81, p = .06), whereas for the 3-concept this assumption was violated (F (1,76) = 22.93, p = .00). To correct the violation of homogeneity, a more stringent significance level was used in the analysis (i.e. 0.01). Furthermore, the assumption of homogeneity of variance-covariance was found to be non-significant (Box's M = 5.34, P > .05), therefore, Wilks' Lambda was selected for interpretation of the results. Finally, as reported previously, the covariates and dependent variables were positively correlated at significance level of 0.01.

According to the MANCOVA result, there was a significant difference between the experimental and control groups on the combined post-test CT conceptual knowledge variables, after controlling for pre-tests: F (3, 73) = 4.17, p = .00, Wilks' Lambda = 0.85, partial η^2 = 0.15. The MANCOVA results for each dependent variable are illustrated in Table 8. A separate examination of the results reveals that there is a significant difference in scores between both groups with regards to the 3-concept items (F (1, 75) = 6.91, p = .01, partial η^2 = 0.08). Nonetheless, for the 2- and 1-concept items, no significant differences were found between groups (F (1, 75) = 3.17, p = .07, partial η^2 = 0.04 for 2-concept and F (1, 75) = 0.91, p = .34, partial η^2 = 0.01 for 1-concept items). For the combined dependent variables, an effect size of 0.15 was found. That is, 15% of the variance in the dependent variables was explained by the independent variables. Both effect size of the combined dependent variables and separate variables were at weak level. While, in 3-concept items, both the mean and adjusted mean score of the experimental group was 1.83 (SE = 0.07), the mean and adjusted mean score of the control group were 1.57 and 1.56 (SE = 0.06), respectively. Furthermore, the mean and adjusted mean score for the experimental group in 2-concept items were 4.27 and 4.18 (SE = 0.15), and for the control group were 3.90 and 3.98 (SE = 0.14), respectively, whereas for the experimental group in 1-concept items both the mean and adjusted mean were 2.83 (SE = 0.06) and for the control group in 1-concept items both were 2.74 (SE = 0.06). Even though in 2-concept and 1-concept items there were no significant differences between the groups, the adjusted means of both groups reveal that students in the experimental group show a higher improvement than the control group. With regards to the 3-concept items, the experimental group significantly outperformed the control group in acquiring CT skills. Overall, these findings suggest the effectiveness of the AutoThinking game compared to the traditional technology-enhanced learning approach in improving students' CT conceptual knowledge.

5.4. CT knowledge gain of students with different prior knowledge

To study the effectiveness of the learning approaches on CT knowledge gain of students with different prior knowledge, students' pre- and post-test scores were used (all ten items in the pre- and post-test). Similar to previous sections, data from 78 students, 36 in the experimental and 42 in the control group, was analyzed. According to Spyridakis and Isakson (1991), students' prior knowledge can affect how students connect prior knowledge to new knowledge. First, according to the mean score of the pre-test, students were divided in two groups: with lower prior knowledge and higher prior knowledge (see Table 9). The students whose result in pre-test was lower than the mean of pre-test were grouped as lower prior knowledge, while those with higher result than the mean were grouped as higher prior knowledge.

As there were slight (insignificant) differences between the pre-test scores of both groups and prior knowledge level of the students, in terms of mean and standard deviation, and such differences may still affect the post-test performance, a two-way ANCOVA was adopted. This would help to obtain more precise statistical data to explain the CT knowledge gain of students, as the pre-test scores were excluded. The ANCOVA used learning approaches and students' prior knowledge (lower and higher) as independent variables, and the posttest and pre-test as the dependent variable and the covariate, respectively. Before conducting the analysis, the basic assumptions of ANCOVA were checked. Accordingly, the regression coefficient indicated no significant interaction between the independent variables and the covariate (F (2,70) = 0.75, p = .48). Additionally, the result of Levene's test was not found to be significant (F (3,74) = 1.01, p = .39), suggesting that between the groups there exists a residual variance homogeneity. As reported before, the covariates and dependent variables were positively correlated at significance level of 0.01.

The result of the two-way ANCOVA indicates, see Table 10, that there was no significant interaction between the independent variables of prior knowledge and learning approaches (F (1,73) = 2.30, p = .13). According to Cohen's (1988), the effect size for this non-significant relationship was found to be small (η^2 = 0.03). Therefore, it would be reasonable if the main effect of the dependent variables were evaluated directly. According to the results, by excluding the pre-test scores, different learning approaches have a significant effect on students' post-test scores (F (1,73) = 12.59, p = .00). The adjusted mean of students in the experimental and control group were 8.90 (SE = 0.18) and 8.06 (SE = 0.17), respectively. This implies that the adaptive game improved students' knowledge gain more than the conventional technology-enhanced learning approach. Furthermore, there was a significant difference on those of the students with different prior knowledge (F (1,73) = 4.53, p = .03). Moreover, the adjusted mean of students with lower and higher prior knowledge is 8.60 (SE = 0.39) and 9.20 (SE

Table 10
Two-way ANCOVA analysis of students' prior knowledge and learning approach.

Groups	Sum of squares	df	Mean square	F	Sig.	Partial η^2
Learning approaches	12.57	1	12.57	12.59	.00*	.15
Prior knowledge	4.52	1	4.52	4.53	.03*	.06
Learning approaches × Prior knowledge	2.29	1	2.29	2.30	.13	.03
Error	72.84	73	.99			

*p < .05.

Table 11
Statistical description of different groups of learning attitude.

Learning approach	Groups	N	Mean	SD
Experimental	Lower prior learning attitude	12	3.70	.47
	Higher prior learning attitude	23	4.54	.32
Control	Lower prior learning attitude	12	3.70	.24
	Higher prior learning attitude	23	4.27	.35

Table 12
Two-way ANCOVA analysis of students' learning attitude.

Groups	Sum of squares	Df	Mean square	F	Sig.	Partial η^2
Learning approaches	.18	1	.18	1.68	.19	.03
Prior learning attitude	.49	1	.49	4.72	.03*	.07
Learning approaches × Prior learning attitude	.20	1	.20	1.88	.17	.02
Error	6.74	65	.10			

*p < .05.

= 0.28) for the group using the adaptive game, respectively, and 7.38 (*SE* = 0.32) and 8.73 (*SE* = 0.22) for the group that learned with the conventional approach, respectively. While the effect size for these significant relationships were found to be small ($\eta^2 = 0.15$ for learning approaches and $\eta^2 = 0.06$ for prior knowledge), further analysis revealed that students with both higher and lower prior knowledge showed higher improvement in knowledge gain using the adaptive game compared to the traditional approach, with lower prior knowledge students appeared to have benefited more than those with higher prior knowledge.

5.5. Learning attitude

To study the effect of different learning approaches (the adaptive educational game versus the conventional technology-enhanced learning) and students prior learning attitude on their attitude toward learning CT, data from the pre- and post-questionnaires collected from both groups of students was used. Data from one student in the experimental group and seven students in the control group were excluded as they did not fully answer the questionnaires. Given the exclusion, data from 70 students was used in the analysis (35 in experimental and 35 in control group). First, according to the mean score of pre-questionnaires, students were divided in two groups: with lower prior learning attitude and higher prior learning attitude (see Table 11).

A two-way ANCOVA was employed with the pre- and post-questionnaire of learning attitude as the covariate and the dependent variable, respectively, along with learning approaches and prior learning attitude (lower and higher) as independent variables. To this end, the basic assumptions of ANCOVA were checked. Accordingly, the regression coefficient indicated no significant interaction between the independent variables and the covariate (F (1,63) = 1.66, p = .20). Additionally, the result of Levene's test was not found to be significant (F (3,66) = 0.89, p = .44), indicating that between the groups there exists a residual variance homogeneity. Moreover, Pearson correlation test on the pre- and post-questionnaire indicated a significant positive correlation between covariate and the dependent variable (r = 0.74; p = .000).

The result of the two-way ANCOVA indicates, see Table 12, that there was no significant interaction between the independent variables, prior learning attitude and learning approaches (F (1,65) = 1.88, p = .17). The effect size for this non-significant relationship was found to be small ($\eta^2 = 0.03$). Therefore, it would be reasonable if the main effect of the dependent variables were evaluated directly. According to the results, while different learning approaches have no significant effect on students' post-questionnaire ratings (F (1,65) = 1.68, p = .19), there is a significant difference on students with different prior learning attitudes (F (1,65) = 4.72, p = .03). Moreover, the adjusted mean of students with lower and higher prior learning attitude is 3.97 (*SE* = 0.12) and 4.38 (*SE* = 0.08) for the group using the adaptive game, respectively, and 3.97 (*SE* = 0.12) and 4.16 (*SE* = 0.07) for the group that learned with the conventional approach, respectively. While the effect size for these relationships were found to be small ($\eta^2 = 0.02$ for learning approaches and $\eta^2 = 0.07$ for prior learning attitude), further analysis revealed that the adaptive game could improve the students' learning attitude better than the conventional approach. More specifically, the adaptive game could improve the learning attitude of students with higher prior learning attitude more than the conventional approach, while students with lower prior learning attitude of both experimental and control groups showed similar improvement in learning attitude.

6. Discussion and conclusions

In this study, an adaptive educational game aimed at promoting students' CT knowledge was introduced (called AutoThinking). Unlike most of the existing educational games, AutoThinking not only promotes both CT skills and conceptual knowledge simultaneously, but also includes adaptivity in both game-play and learning process. To evaluate the effectiveness of the adaptive game compared to a conventional technology-enhanced learning approach (i.e., a lesson using a PowerPoint presentation with Multimedia delivered by a teacher), an experimental study on elementary school students was conducted. More specifically, it was investigated 1) whether the adaptive game improves students' skills and conceptual knowledge in CT better than the traditional approach, 2) the effect of different learning approaches (the adaptive game and the traditional approach) on CT knowledge gain of the students with different prior knowledge (low and high), and 3) the effect of students different prior learning attitude (low and high) and learning approaches on their learning attitude toward CT.

Our findings reveal that AutoThinking improved students' CT skills better than the conventional approach. That is, the experimental group significantly outperformed the control group in acquiring all four CT skills simultaneously (algorithmic thinking, pattern recognition, debugging and simulation). More explicitly, this result indicates that students who learned with the adaptive game could show a superior performance in acquiring CT skills compared to the control group, especially in pattern recognition and debugging skill. That is, the game improved students' ability to develop repeatable (and generalizable) strategies, monitor their solution algorithm, and detect potential errors in its logic. As it is apparent, the conventional technology-enhanced learning approach could not trigger this skill and students in the control group mostly could not debug their solutions and find their possible mistakes and errors in their logic (compared to the adaptive game). One reason for this result is the individual support provided by the game. Students could, in a timely manner and according to their level of skills, receive feedback, hints, and learning materials, helping them to recognize patterns and create re-useable strategies, as well as monitor their solutions and detect CT-related mistakes in their logic during the game-play. Besides, the adaptivity feature of the game during the game-play could be one reason for the students' retention, motivation, and success.

Similarly, our findings show that AutoThinking improved students' conceptual knowledge of CT better than the conventional approach. That is, the experimental group significantly outperformed the control group in acquiring all three CT concepts together (sequence, conditional logic, and loop logic). More explicitly, this result shows that students who learned with the adaptive game could show a superior performance in acquiring CT conceptual knowledge compared to the control group, especially in conditional concept. That is, the game improved students' ability in decision-making based on certain conditions that support the result of multiple outcomes of their developed algorithms. As it is apparent, the conventional technology-enhanced learning approach could not trigger this concept properly and students in the control group mostly appeared to not have improved enough (as good as the experimental group) in conditional logic that allows them to automatically make decisions based on a condition or action that occurred. One crucial reason for this result is the individual support provided by the game. Students could, in a timely manner and according to their level of knowledge, receive feedback, hints, and learning materials related to conditional logic which help them to find their mistakes in using conditional logic (improving their decision-making ability based on different conditions during the game-play). Even though it might not be fully correct to compare our findings with previous research due to several reasons—e.g., different experimental design, sample size, and the intervention type—our result generally is in line with findings from previous studies on educational games aimed at developing students' CT (e.g., Eagle & Barnes, 2009; Horn et al., 2016; Zhao & Shute, 2019). Needless to mention, this result is consistent with findings of previous research on the positive effect of (adaptive) educational games in general (e.g., Hooshyar et al., 2018a; Hwang, Sung, Hung, Huang, & Tsai, 2012; Shute, Rieber, & Van Eck, 2011; Sung & Hwang, 2013).

Although there are a few studies reporting the effectiveness of an educational game by comparing educational games with traditional technology-enhanced learning approaches for improving students' CT, multiple research studies have reported the effectiveness of educational games in terms of developing students' CT using a one-group pretest-posttest design (e.g., Zhao & Shute, 2019). However, there has not been an experimental study, to the best of our knowledge, reporting the effectiveness of an educational game in terms of the learning achievement of students in CT (including both conceptual knowledge and skills). While further research is required to find the main reason behind the significant improvement in the CT knowledge (both skills and concepts) of students who learned with AutoThinking, one probable reason could be the adaptivity feature of the game that both during the game-play and learning process supports individual needs. To this end, several researchers have also reported that adaptivity and personalization in educational games facilitate reaching their full educational potential (e.g., Hooshyar et al., 2018a; Hwang, Kuo, Yin, & Chuang, 2010; Kickmeier-Rust et al., 2011; Sampayo-Vargas, Cope, He, & Byrne, 2013).

With regard to the effect of different learning approaches on the CT knowledge gain of the students with different prior knowledge level, our findings show that while students with both higher and lower prior knowledge showed improvement in knowledge gain using the adaptive game compared to the traditional approach, students with lower prior knowledge appeared to have benefited more than those with higher prior knowledge. This result is consistent with findings of previous research on the positive effect of game-based approaches on students with low prior knowledge compared to conventional technology enhanced-learning approaches in other fields, such as mathematics (McLaren, Adams, Mayer, & Forlizzi, 2017).

Finally, concerning the effect of different learning approaches on the learning attitude of the students with different prior learning attitudes, our findings reveal that the adaptive game could improve the students' learning attitude more than the conventional approach. More explicitly, the adaptive game could improve the learning attitude of students with higher prior learning attitude better than the conventional approach, while students with lower prior learning attitude of both experimental and control groups showed similar improvement in learning attitude. This shows that even though CT has recently been added to the formal curricula of school students (it is a fairly new subject), and several researchers have found that many students show negative behavior toward learning computer science-based subjects in early education (e.g., Yardi & Bruckman, 2007), our game could successfully improve students' learning attitude toward CT. One reason for this improvement could be that elementary school students often prefer to learn through games rather than using traditional approaches. Our findings with regard to learning attitude toward CT are in line with those reported by other researchers (in programming for example; Sharma, Papavlasopoulou, & Giannakos, 2019).

Note that as the participants of the study were aware that they were being measured by different means (e.g., tests and questionnaires), the Hawthorne effect should be considered. The Hawthorne effect has been interpreted as an increase in motivation and productivity of a group simply because they are being studied (French, 1953; Mayo, 2004). In the present study, to avoid the Hawthorne effect and some other similar effects (e.g., John Henry effect), the experimental group was not treated differently from the control group, the groups were not informed that there is another group of students participating in the experiment, before the experiment the classes were randomly assigned to the experimental and control group, both groups learned in the same environment at different times, and all participants were taught by the same teacher. As a future work, it would be interesting to investigate more precisely the adaptivity feature of the game and the possible effect of different adaptivity features in the game on students' learning achievement, motivation, learning attitude, and technology acceptance.

CRediT authorship contribution statement

Danial Hooshyar: Conceptualization, Data curation, Formal analysis, Methodology, Supervision, Visualization, Writing - review & editing. **Liina Malva:** Conceptualization, Data curation, Formal analysis, Methodology, Supervision, Writing - review & editing. **Yeongwook Yang:** Conceptualization, Data curation, Formal analysis, Writing - review & editing. **Margus Pedaste:** Data curation, Formal analysis, Writing - review & editing. **Minhong Wang:** Data curation, Formal analysis, Writing - review & editing. **Heuiseok Lim:** Data curation, Formal analysis, Writing - review & editing.

Acknowledgements

This research was supported by the University of Tartu ASTRA Project PER ASPERA, financed by the European Regional Development Fund; the MSIT (Ministry of Science and ICT), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2020-2018-0-01405) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation); and the Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00368, A Neural-Symbolic Model for Knowledge Acquisition and Inference Techniques).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.chb.2020.106575.

References

Allen, P., & Bennett, K. (2008). *SPSS for the health & behavioural sciences*. Melbourne, Australia: Thomson.
Angeli, C., & Giannakos, M. (2020). Computational thinking education: Issues and challenges. *Computers in Human Behavior, 105*, 106185.
Angeli, C., & Valanides, N. (2020). Developing young children's computational thinking with educational robotics: An interaction effect between gender and scaffolding strategy. *Computers in Human Behavior, 105*, 105954.
Ater-Kranov, A., Bryant, R., Orr, G., Wallace, S., & Zhang, M. (2010). Developing a community definition and teaching modules for computational thinking: Accomplishments and challenges. In *Proceedings of the 2010 ACM conference on information technology education* (pp. 143–148). ACM.
Ayman, R., Sharaf, N., Ahmed, G., & Abdennadher, S. (2018, November). MiniColon; teaching kids computational thinking using an interactive serious game. *Lecture Notes in Computer Science, 11243*, 79–90 (Springer).
Barr, D., Harrison, J., & Conery, L. (2011). Computational thinking: A digital age skill for everyone. *Learning and Leading with Technology, 38*(6), 20–23.
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.
Blum, L., & Cortina, T. J. (2007). CS4HS: An outreach program for high school CS teachers. *ACM SIGCSE Bulletin, 39*(1), 19–23.
Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American educational research association* (Vol. 1), 25.
Brown, N. C., Sentance, S., Crick, T., & Humphreys, S. (2014). *Restart: The resurgence of computer science in UK schools* (Vol. 14). ACM Transactions on Computing Education (TOCE), 2.
Carlborg, N., Tyren, M., Heath, C., & Eriksson, E. (2019). The scope of autonomy when teaching computational thinking in primary school. *International Journal of Child-Computer Interaction, 21*, 130–139.
Cohen, J. (1988). *Statistical power analysis for the behavioral sciences* (2nd ed.). Hillsdale, NJ: Erlbaum.
Denning, P. J. (2009). The profession of IT beyond computational thinking. *Communications of the ACM, 52*(6), 28–30.
DiSessa, A. A. (2000). *Changing minds*. MIT press.

Eagle, M., & Barnes, T. (2009). Experimental evaluation of an educational game for improved learning in introductory computing. *ACM SIGCSE Bulletin, 41*(1), 321–325.

El Mawas, N., Hooshyar, D., & Yang, Y. (2020). Investigating the learning impact of AutoThinking educational game on adults: A case study of France. In *, Vol. 2. Proceedings of the CSEDU* (pp. 188–196).

Esper, S., Foster, S. R., Griswold, W. G., Herrera, C., & Snyder, W. (2014, November). CodeSpells: Bridging educational language features with industry-standard languages. In *Proceedings of the 14th koli calling international conference on computing education research* (pp. 5–14).

Färnqvist, T., Heintz, F., Lambrix, P., Mannila, L., & Wang, C. (2016, February). Supporting active learning by introducing an interactive teaching tool in a data structures and algorithms course. In *Proceedings of the 47th ACM technical symposium on computing science education* (pp. 663–668).

Field, A. (2009). *Discovering statistics using SPSS: (and sex and drugs and rock'n'roll)*. Sage.

Fraser, N. (2013). *Blockly: A visual programming editor*. Retrieved from https://developers.google.com/blockly/.

French, J. R. (1953). Experiments in field settings. In L. Festinger, & D. Katz (Eds.), *Research methods in the behavioral sciences* (pp. 95–135). New York: Holt, Rinehart & Winston.

Giannakoulas, A., & Xinogalos, S. (2018). A pilot study on the effectiveness and acceptance of an educational game for teaching programming concepts to primary school students. *Education and Information Technologies, 23*(5), 2029–2052.

González, M. R. (2015, July). Computational thinking test: Design guidelines and content validation. In *Proceedings of EDULEARN15 conference* (pp. 2436–2444).

Gouws, L. A., Bradshaw, K., & Wentworth, P. (2013, July). Computational thinking in educational activities: An evaluation of the educational game light-bot. In *Proceedings of the 18th ACM conference on Innovation and technology in computer science education* (pp. 10–15).

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43.

Guzdial, M. (2008). Education paving the way for computational thinking. *Communications of the ACM, 51*(8), 25–27.

Haddad, R. J., & Kalaani, Y. (2015, March). Can computational thinking predict academic performance?. In *Proceedings of the 2015 IEEE integrated STEM education conference* (pp. 225–229). IEEE.

Harvey, B., & Mönig, J. (2010). Bringing "no ceiling" to scratch: Can one language serve kids and computer scientists. In *Proceedings of the constructionism 2010 conference* (pp. 1–10).

Hooshyar, D., Lim, H., Pedaste, M., Yang, K., Fathi, M., & Yang, Y. (2019). December). AutoThinking: An adaptive computational thinking game. In *International Conference on Innovative Technologies and Learning* (pp. 381–391). Cham: Springer.

Hooshyar, D., Yousefi, M., & Lim, H. (2018). Data-driven approaches to game player modeling: a systematic literature review. *ACM Computing Surveys (CSUR), 50*(6), 1–19.

Hooshyar, D., Yousefi, M., Wang, M., & Lim, H. (2018). A data-driven procedural-content-generation approach for educational games. *Journal of Computer Assisted Learning, 34*(6), 731–739.

Horn, B., Clark, C., Strom, O., Chao, H., Stahl, A. J., Harteveld, C., et al. (2016, February). Design insights into the creation and evaluation of a computer science educational game. In *Proceedings of the 47th ACM technical symposium on computing science education* (pp. 576–581).

Hwang, G.-J., & Chang, H.-F. (2011). A formative assessment-based mobile learning approach to improving the learning attitudes and achievements of students. *Computers & Education, 56*(4), 1023–1031.

Hwang, G.-J., Chiu, L.-Y., & Chen, C.-H. (2015). A contextual game-based learning approach to improving students' inquiry-based learning performance in social studies courses. *Computers & Education, 81*, 13–25.

Hwang, G.-J., Kuo, F.-R., Yin, P.-Y., & Chuang, K.-H. (2010). A heuristic algorithm for planning personalized learning paths for context-aware ubiquitous learning. *Computers & Education, 54*(2), 404–415.

Hwang, G.-J., Sung, H.-Y., Hung, C.-M., Huang, I., & Tsai, C.-C. (2012). Development of a personalized educational computer game based on students' learning styles. *Educational Technology Research & Development, 60*(4), 623–638.

Kazimoglu, C., Kiernan, M., Bacon, L., & Mackinnon, L. (2012a). A serious game for developing computational thinking and learning introductory computer programming. *Procedia-Social and Behavioral Sciences, 47*, 1991–1999.

Kazimoglu, C., Kiernan, M., Bacon, L., & MacKinnon, L. (2012b). Learning programming at the computational thinking level via digital game-play. *Procedia Computer Science, 9*, 522–531.

Kazimoglu, C., Kiernan, M., Bacon, L., & MacKinnon, L. (2013). Understanding computational thinking before programming: Developing guidelines for the design of games to learn introductory programming through game-play. In P. Felicia (Ed.), *Developments in current game-based learning design and deployment* (pp. 316–338). IGI Global.

Kickmeier-Rust, M. D., Mattheiss, E., Steiner, C., & Albert, D. (2011). A psycho-pedagogical framework for multi-adaptive educational games. *International Journal of Game-Based Learning, 1*(1), 45–58.

Ku, O., Hou, C.-C., & Chen, S. Y. (2016). Incorporating customization and personalization into game-based learning: A cognitive style perspective. *Computers in Human Behavior, 65*, 359–368.

Kuruvada, P., Asamoah, D., Dalal, N., & Kak, S. (2010, November). Learning computational thinking from rapid digital game creation. In *Proceedings of the 2nd annual conference on theoretical and applied computer science* (pp. 31–36).

Land, S. M. (2000). Cognitive requirements for learning with open-ended learning environments. *Educational Technology Research & Development, 48*(3), 61–78.

Malva, L., Hooshyar, D., Yang, Y., & Pedaste, M. (2020, July). Engaging Estonian primary school children in computational thinking through adaptive educational games: A qualitative study. In *Proceedings of the IEEE 20th international conference on advanced learning technologies* (pp. 188–190).

Mayo, E. (2004). *The human problems of an industrial civilization*. London and New York: Routledge.

McLaren, B. M., Adams, D. M., Mayer, R. E., & Forlizzi, J. (2017). A computer-based game that promotes mathematics learning more than a conventional approach. *International Journal of Game-Based Learning, 7*(1), 36–56.

Meerbaum-Salant, O., Armoni, M., & Ben-Ari, M. (2011, June). Habits of programming in scratch. In *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education* (pp. 168–172).

National Research Council. (2010). *Report of a workshop on the scope and nature of computational thinking*. National Academies Press.

Palts, T., & Pedaste, M. (2020). A model for developing computational thinking skills. *Informatics in Education, 19*(1), 113–128.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. New York, NY: Basic Books.

Papert, S. (1996). *Computers in the classroom: Agents of change* (Vol. 27). The Washington Post Education Review.

Peirce, N., Conlan, O., & Wade, V. (2008, November). Adaptive educational games: Providing non-invasive personalised learning experiences. In *Proceedings of the 2008 second IEEE international conference on digital game and intelligent toy enhanced learning* (pp. 28–35). IEEE.

Perković, L., Settle, A., Hwang, S., & Jones, J. (2010, June). A framework for computational thinking across the curriculum. In *Proceedings of the fifteenth annual conference on Innovation and technology in computer science education* (pp. 123–127).

Pontes, H. P., Duarte, J. B. F., & Pinheiro, P. R. (2020). An educational game to teach numbers in Brazilian Sign Language while having fun. *Computers in Human Behavior, 107*, 105825.

Repenning, A. (n.d.). Computational Thinking ≠ Programming. SI Magazine. Swiss Informatics Society. Retrieved from https://magazine.swissinformatics.org/en/computational-thinking-%e2%89%a0-programming/.

Repenning, A., Webb, D., & Ioannidou, A. (2010, March). Scalable game design and the development of a checklist for getting computational thinking into public schools. In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 265–269).

Repenning, A., Webb, D. C., Koh, K. H., Nickerson, H., Miller, S. B., Brand, C., et al. (2015). *Scalable game design: A strategy to bring systemic computer science education to schools through game design and simulation creation* (Vol. 15). ACM Transactions on Computing Education (TOCE), 2.

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

Román-González, M., Pérez-González, J.-C., & Jiménez-Fernández, C. (2017). Which cognitive abilities underlie computational thinking? Criterion validity of the computational thinking test. *Computers in Human Behavior, 72*, 678–691.

Saines, G., Erickson, S., & Winter, N. (2013). *Codecombat*. Silicon Valley, CA: CodeCombat.

Sampayo-Vargas, S., Cope, C. J., He, Z., & Byrne, G. J. (2013). The effectiveness of adaptive difficulty adjustments on students' motivation and learning in an educational computer game. *Computers & Education, 69*, 452–462.

Selby, C., & Woollard, J. (2013). Computational thinking: The developing definition. In *Paper presented at the 18th annual conference on innovation and technology in computer science education*.

Sharma, K., Papavlasopoulou, S., & Giannakos, M. (2019). Coding games and robots to enhance computational thinking: How collaboration and engagement moderate children's attitudes? *International Journal of Child-Computer Interaction, 21*, 65–76.

Shute, V. J., Rieber, L., & Van Eck, R. (2011). Games... and... learning. In R. Reiser, & J. Dempsey (Eds.), *Trends and issues in instructional design and technology* (3rd ed., pp. 321–332). Upper Saddle River, NJ: Pearson Education.

Spyridakis, J. H., & Isakson, C. S. (1991). Hypertext: A new tool and its effect on audience comprehension. In *Proceedings of the IPCC 91 engineered communication* (pp. 37–44).

Sung, H. Y., & Hwang, G. J. (2013). A collaborative game-based learning approach to improving students' learning performance in science courses. *Computers & Education, 63*, 43–51.

Tabachnick, B. G., Fidell, L. S., & Ullman, J. B. (2007). *Using multivariate statistics*. Boston, MA: Pearson.

Tsarava, K., Moeller, K., Pinkwart, N., Butz, M., Trautwein, U., & Ninaus, M. (2017, October). Training computational thinking: Game-based unplugged and plugged-in activities in primary school. In *Proceedings of the European conference on games based learning* (pp. 687–695).

Vandercruysse, S., ter Vrugte, J., de Jong, T., Wouters, P., van Oostendorp, H., Verschaffel, L., et al. (2016). The effectiveness of a math game: The impact of integrating conceptual clarification as support. *Computers in Human Behavior, 64*, 21–33.

Weintrop, D., & Wilensky, U. (2012). RoboBuilder: A program-to-play constructionist video game. In *Proceedings of the constructionism 2012 conference* (Athens, Greece).

Wilensky, U. (2001, August). Modeling nature's emergent patterns with multi-agent languages. In *Proceedings of EuroLogo* (pp. 1–6).

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

Wing, J. (2011). Computational thinking—what and why. *The Link Magazine, 6*, 20–23. Spring.

Wing, J. M. (2014). *Computational thinking benefits society*. Retrieved from http://socialissues.cs.toronto.edu.

Xie, H., Chu, H. C., Hwang, G. J., & Wang, C. C. (2019). Trends and development in technology-enhanced adaptive/personalized learning: A systematic review of journal publications from 2007 to 2017. *Computers & Education, 140*, 103599.

Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). *Computational thinking in elementary and secondary teacher education* (Vol. 14). ACM Transactions on Computing Education (TOCE), 1.

Yardi, S., & Bruckman, A. (2007, September). What is computing? Bridging the gap between teenagers' perceptions and graduate students' experiences. In *Proceedings of the third international workshop on Computing education research* (pp. 39–50).

Zhao, W., & Shute, V. J. (2019). Can playing a video game foster computational thinking skills? *Computers & Education, 141*, 103633.

Zumbach, J., Rammerstorfer, L., & Deibl, I. (2020). Cognitive and metacognitive support in learning with a serious game about demographic change. *Computers in Human Behavior, 103*, 120–129.