

Article

# Assessing Algorithmic Thinking Skills in Relation to Age in Early Childhood STEM Education

Kalliopi Kanaki \* and Michail Kalogiannakis \* 

Department of Preschool Education, University of Crete, 74100 Rethymno, Greece

\* Correspondence: kalkanaki@uoc.gr (K.K.); mkalogian@uoc.gr (M.K.)

**Abstract:** In the modern digital era, intensive efforts are made to inject computational thinking (CT) across science, technology, engineering, and mathematics (STEM) fields, aiming at formulating a well-trained citizenry and workforce capable of confronting intricate problems that would not be solvable unless exercising CT skills. Focusing on contributing to the research area of CT assessment in the first two years of primary school, we investigated the correlation of algorithmic thinking skills, as a fundamental CT competency, with students' age in early childhood settings. This article reports a relevant research study, which we implemented under the umbrella of quantitative methodology, employing an innovative assessment tool we constructed for serving the needs of our study. The research was conducted within the context of the environmental study course, adding to the efforts of infusing CT into STEM fields. The study results shed light on the correlation between algorithmic thinking skills and age in early childhood, revealing that age is a predictor factor for algorithmic thinking and, therefore, for CT.

**Keywords:** computational thinking; algorithmic thinking; environmental study; early education; age



**Citation:** Kanaki, K.; Kalogiannakis, M. Assessing Algorithmic Thinking Skills in Relation to Age in Early Childhood STEM Education. *Educ. Sci.* **2022**, *12*, 380. <https://doi.org/10.3390/educsci12060380>

Academic Editors: João Piedade and James Albright

Received: 23 April 2022

Accepted: 27 May 2022

Published: 31 May 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

CT is being embraced worldwide as a 21st-century core skill set that upcoming generations should cultivate [1–3] since it is related to competencies beneficial and applicable in everyday life [2,4]. Despite the considerable attention CT has been receiving from the educational and the scientific community [5] since Wing [4] coined the term, there is neither a clear-cut definition [3], nor an agreement on its constituent skills [6]. Surprisingly, in spite of this lack of consensus, CT has already been included in several countries' curricula for K-12 education [1,7], with the up-to-date dominant tendency to move away from the cultivation of CT as an isolated curriculum domain. On the contrary, the attention is focused on its integration into existing disciplines, with the emphasis placed on STEM fields [7–9].

Most of the relevant endeavors are concentrated in the so-called core science fields, such as physics [10] and mathematics [1,11], leaving a research gap in the case of the synergistic cultivation and assessment of CT and environmental science. Having detected this research gap and being impelled by the persistent need to (a) formulate an environmentally aware citizenry [12], and (b) cultivate CT as an essential stepping-stone in the modern digital life [13,14], we are engaged in bringing CT in the environmental science field in compulsory education.

However, since the cognitive capacity of the students varies depending on their age, the methods, contents, and learning strategies for teaching and evaluating CT should be adapted accordingly [15]. Towards this end, we conducted a research study with the objective of investigating students' algorithmic thinking—as a fundamental CT skill—within the context of environmental science—as a STEM branch. Our study focused on examining students' algorithmic thinking skills in relation to their age in the first two years of primary school. Thus, the research question that drives this article is: “Are algorithmic

thinking skills exercised in environmental science classes related to students' age in the early primary school years?"

In order to answer the research question, we set the hypothesis that: "Algorithmic thinking skills are not associated to students' age in the early primary school years". Reviewing previously published literature pertinent to the research problem, we found out that until now, the influence of age on the development of CT skills in primary education is examined and confirmed in samples of wide age ranges [16,17]. Elaborating on the rationality of our study, we should point out that, the case of substantiating the development of CT skills in wide age ranges does not imply that it is also observable in a narrow age range, especially when it concerns the beginning of schooling. The need of investigating the influence of age on the development of CT skills in the sensitive years of early education appertains to specifying whether there is need of constructing different tools for cultivating CT in the first two years of schooling, or whether the same tools might be developmentally appropriate.

To test the hypothesis set, we designed and implemented an age-appropriate digital platform PhysGramming (an acronym derived from Physical Science Programming) of constructivist nature that provides students the opportunity to create their own digital games and, then, play with them. PhysGramming's character is formulated according to the widespread belief that coding ranks among the most efficient means of exercising CT skills [18]. Being aware of the difficulty of employing text-based programming languages [18], especially in early education, PhysGramming adopts a hybrid schema of text-based and visual programming techniques, with the emphasis laid on object-orientation [19]. The game-based character of PhysGramming captures students' attention [20–22], while its multidisciplinary aspect manifests students' involvement with CT activities in environmental study settings.

The relevant research study was conducted in Heraklion city, on the island of Crete, in the school year 2018–2019, following a robust ethical protocol [23,24] and adopting the quantitative research methodology. The research process lasted six months, from January to June 2019. Each intervention lasted two instructional hours with the objective of investigating the effectiveness of the proposed assessment framework and the relation of fundamental CT skills, such as algorithmic thinking and abstraction, with factors such as gender, age etc. [25]. The part of the interventions that focused on the objective discussed in this article lasted half an hour. In all, 435 first and second graders participated in the research process. At this point, we should clarify that, in Greece, six is usually the primary school starting age.

Regarding the methods of analysis applied to the data obtained in determining whether the hypothesis set is true or false, we implemented chi-square test, and calculated *p*-value and odds ratio. We also employed the ordinal logistic regression analysis method to model the relationship between the variables under investigation. Finally, we used the machine learning method [26] to predict the algorithmic thinking levels of similar populations.

Answering the research question contributes to assessing algorithmic thinking via developmentally appropriate techniques in the early years within the context of STEM courses. In a broader context, our study is important to appropriately cultivating CT as an attitude universally applicable [2,4], which is not confined to programming and computer science [27]. On the contrary, it applies to approaching and seeking the solutions to various problems in a way that a computer scientist would [4,28].

## 2. Theoretical Framework

The recognized importance of cultivating CT in PK-12 paved the way for the rapid development of relevant educational approaches over the past decade [27]. Nonetheless, the subject of evaluating CT remains open, as a research challenge, demanding scholars' attention urgently [29–31]. Otherwise, the structure of CT will be threatened with disappearing, although it deserves very serious consideration. Indeed, its adoption by

contemporary K-12 curricula would be problematic without establishing valid and reliable assessment methods first [32], since they are pivotal for evaluating students' prospective learning progress and, therefore, the efficacy of proposed curricular programs [33].

Towards this end, we introduce a multidisciplinary, game-based assessment tool, the backbone of which is PhysGramming, focusing on evaluating essential CT skills—such as algorithmic thinking—in early childhood. Apart from elaborating on the assessment aspect of PhysGramming, we have paid extra attention to designing a developmentally appropriate framework that brings early learners in contact with object-oriented programming in STEM settings.

### 2.1. *The Environmental Study Course*

Nowadays, environmental education becomes more and more popular at all levels of education, since it investigates the Earth's systems, human's effect on these, and public health matters. It provides a multidisciplinary comprehension of the interdependence between people, the environment, and the economy [34]. It captures the interest of children and adults who are interested in investigating the world around them, its prospects, and their place within it [34].

In Greece, the environmental study course in the first and second primary school grade has an interdisciplinary character, since it is an undivided and integrated field of learning, where structural elements from humanities and social sciences, as well as natural sciences, are intertwined. Its main purpose is to help students, through collaborative exploration, to gain a fundamental and essential conceptual background in different areas of science, as well as to acquire a valuable body of information, which will eventually be transformed into meaningful knowledge. The content of the course is organized in thematic units related to modern life. Thus, by its nature, the course is connected to the community and the life in it and, therefore, supports authentic learning, via several topics such as: social organization, the individual and their needs, space and interdependence with people's lives, rights and obligations of members of a community, the municipality, the geographical regions of Greece, transportation (in relation to space and human needs), the natural environment, ecosystems and their protection, time in everyday life, the economy and its relation to space and the organization of societies, life and needs of people, communication and information, the culture of Greeks and other peoples, modules from Physics (energy, sound, etc.).

### 2.2. *Computational Thinking*

Digital culture has permeated contemporary societies [35]. At the same time, the use of computational methods is not limited to computer science [36–38]. Many other sciences have already established the exploitation of computational techniques in order to introduce their concepts and elaborate information [38,39]. The fact that computer science applications engulf modern societies and support the evolutionary steps of other sciences, raises the impelling need of introducing computer science in K-12 [39,40].

Meanwhile, since Wing [4] has reintroduced the term, a long-lasting wave propels CT's introduction in K-12 as a core element of computer science, but also proposes its exploitation in many other scientific domains, imbuing it with interdisciplinary value [39]. Despite the vagueness of its definition, its assessment, and its core components, a considerable number of countries have already moved towards the formulation of frameworks that will bring CT to schools, starting from kindergarten [39,41,42].

Early education has traditionally laid the foundation for cultivating reading, writing, and mathematics. The updated requirements of the modern world introduce the significance of facing CT as a fundamental skill set that every child must exercise [43]. Nonetheless, CT is not instinctive to humankind and requires systematic and developmentally appropriate training and guidance. Thus, the challenge is to establish educational settings that will facilitate children to become equally comfortable with CT as they are with other fundamental fields of early education [40,43].

### 2.2.1. Computational Thinking and Programming

Since CT draws on fundamental computer science concepts [4], it is beyond doubt that programming exposes students to CT [18,44]. Nevertheless, although CT skills become alive in a programming context, they could also be projected onto a plethora of tasks irrelevant to programming [44,45]. That is to say, CT skills are certainly required for programming, but they also might be employed in other fields not related to programming. Thus, CT is a broader term than programming [32]. It surpasses the narrow perspective of learning programming syntax, leading to the development of capabilities such as thinking conceptually and solving problems at multiple levels of abstraction [4].

Developing an interdisciplinary framework of establishing CT in compulsory education in no computer science settings would support all the parties involved to (a) understand complex systems, (b) be inventive with computational representations, (c) propose solutions that exploit computational resources, (d) be engaged in collective actions aiming at giving meaning to data, (e) canvas potential consequences of actions [13].

### 2.2.2. Thinking Computationally in Environmental Science

Policymakers face CT as an essential part of STEM disciplines in K-12 and propose the adoption of interdisciplinary approaches that promote the integration of CT into STEM classrooms, supporting students across K-12 to cultivate knowledge and skills addressed in both CT and STEM subjects [46]. Efforts towards this end have been considerably boosted with the inclusion of CT practice in the Next Generation Science Standards [47].

Despite the lack of efforts toward the synergistic cultivation and assessment of CT and environmental science in K-12, there are several scientific attempts seeking to formulate a CT framework from an environmental science perspective, exploiting multidisciplinary. These attempts would be difficult, or even impossible, to accomplish if CT was not employed. For example, CT supports constructing water management strategies, depending on each community's water usage and needs [13].

### 2.2.3. Algorithmic Thinking

An algorithm is a problem-solving approach that consists of a finite number of well-defined and ordered steps that are executable in a finite amount of time. Algorithmic thinking is an individual's ability to construct new algorithms aiming at solving a given problem [48]. It is inextricably related to the abstraction ability, which is the essence of CT [2,4]. It is also determined as a pool of skills that foster algorithms' construction and comprehension, namely problem analysis, problem specification, algorithm design, and algorithm optimization [48].

Scholars underline the importance of exercising algorithmic thinking and developing problem-solving skills even from early childhood aiming at efficiently cultivating CT and developing positive dispositions against technical professions [49].

Algorithmic thinking should not be seen just through the prism of computer science, since it is applicable in everyday life [50], providing a solid basis for comprehending how to attain goals by formulating a series of explicit steps [51]. In fact, an algorithmic world-view is very important in the sense of supporting the implementation of various essential daily activities by following simple and well-defined steps [52]. Since a problem-solving process might result in more than one adequate approach, that is to say, algorithms, algorithmic thinking skills entail, among others, the ability to detect the optimal sequence of steps toward predefined objectives [51].

### 2.2.4. Assessing Algorithmic Thinking

In recent years, CT competitions have been established all over the world for K-12 students. The relevant tests focus on algorithms. In some cases, they are implemented using a computer, so students' programming skills can be tested, among others. An example of such a competition is Bebras [53], which started in Lithuania in 2003, and effective supervision by teachers today has been adopted by many European countries,

including Greece. Bebras aims to promote the interest and excellence of compulsory education students around the world in the field of computer science, from the perspective of CT [53]. Every year, the competition introduces new tests related to “real life” problems that, in order to be solved, require students to exploit and demonstrate their CT skills [32]. Issues to consider in such competitions are the different levels of students’ familiarity with digital technology, as well as schools’ ability to support mass participation, by providing computer to all the students and by ensuring effective supervision by teachers [54]. An additional requirement is the independence of solving the tests from computers’ hardware and software, as well as from programming languages [53].

Other competitions are paper-based, with assignments that are answered quickly (multiple choice or right/wrong questions). They do not require prior knowledge and their assignments are graded, from very simple questions to most demanding ones. They are also designed to be easily managed by teachers and schools, even when students’ participation is massive. The difficulty in such competitions is that, since they focus on algorithms and algorithmic thinking, it is not always easy to find questions that will be algorithmic in nature and will not require prior programming knowledge [54,55].

Studying the relevant literature base, we also find several research approaches regarding the assessment of CT competencies, such as algorithmic thinking. Examining the methods and tools they employ, we notice that robotics, gameplay, reasoning everyday events, and programming environments stand out as the most prevalent [29,56–59].

Conceptualizing CT according to Computer Science Teacher Association’s standards, a CT assessment instrument is proposed for fifth graders. The included items are classified in two types of CT activities, namely, coding in robotics and reasoning of daily events. The instrument has been administered in a primary school where a novel humanoid robotics curriculum is applied in the fifth grade. A relevant research study conducted reveals that the proposed instrument could assess the development of a set of CT skills, with algorithmic thinking being one of them [56].

Fairy Assessment proposes the use of Alice by high school students, who are asked to code parts of an already designed algorithm. The goal is to complete specific tasks, in order to reveal students’ algorithmic thinking levels, as well as their abstraction ability and programming knowledge [57].

The FACT (an acronym derived from Foundations for Advancing Computational Thinking) is recommended for high school students and is based on the use of the Scratch programming environment. It measures, among others, the development of algorithmic thinking skills and the transfer of these skills from block-based to text-based programming (two different syntaxes are involved: one that resembles Pascal and one that resembles Java). In addition, it focuses on non-cognitive aspects, such as students’ beliefs and perceptions about Informatics in general and computer programming in particular [29].

The CT learning game Zoombinis is proposed as a novel form of assessing CT skills, such as problem decomposition, pattern recognition, abstraction and algorithm design [58]. It provides a suite of twelve puzzles, each with four levels, that allow students (ages eight and above) to exercise their problem-solving abilities and CT competencies via scaffolded activities [59].

### 2.3. Game-Based Learning

Neuroscience research reveals that the most developmentally appropriate educational practice in the early years is via playing [21,22]. Play-based activities intrigue young children, maintain their attention and engagement in learning, elicit feedback, and facilitate skills consolidation [20–22,60].

Moreover, in the contemporary technological era, the learning needs and demands of the digital natives’ generation challenge the traditional model of classroom-based education. Students who are raised using technology have less patience to sit in classrooms and listen to their teachers. Alternative models of education such as game-based learning are slowly evolving [61].

As far as algorithmic thinking is concerned, research reveals that the majority of students are not able to develop it efficiently if they operate in traditional learning environments [62]. On the contrary, the construction of digital games is influential in the sense of students' learning motivation and engagement in the educational process, ending up in the improvement of their educational achievements [62].

### Jigsaw Puzzles

It is not new that people derive pleasure from reconstructing an image cut into pieces [63,64]. Besides its fun aspect, this activity finds application in several areas of everyday life [64,65]. Moreover, it is related to numerous scientific fields and professional sectors, including biology, archeology, image processing [66,67], and voice communication security [68].

The topic of studying jigsaw puzzles and investigating their potential algorithmic solutions has already concentrated several research efforts [63,66]. In fact, the topic pertains to the sphere of challenging mathematical and engineering problems, captivating the attention of computer scientists, mathematicians and engineers [64]. The engagement of researchers and scientists with exploring algorithms to solve jigsaw puzzles, formulates fertile ground for the standpoint that algorithmic thinking is a prerequisite for solving jigsaw puzzles. Within this context, the ability of solving jigsaw puzzles could comprise a criterion for evaluating algorithmic thinking skills.

## 3. Materials and Methods

The lack of consensus in terms of CT's unambiguous definition and constituent elements makes the development of valid and reliable instruments for measuring primary students' CT skills a challenging venture [69]. On the other hand, experts agree that employing playful programming environments empower CT skills [49].

The innovative idea of introducing object-oriented programming concepts in early childhood education could not be served by the existing tools. This is the very reason we provide the educational community with PhysGramming [19], which not only supports multidisciplinary assessment but also gives prominence to the stance that the first contact with programming is better to be attempted through the object-oriented paradigm [19,70,71].

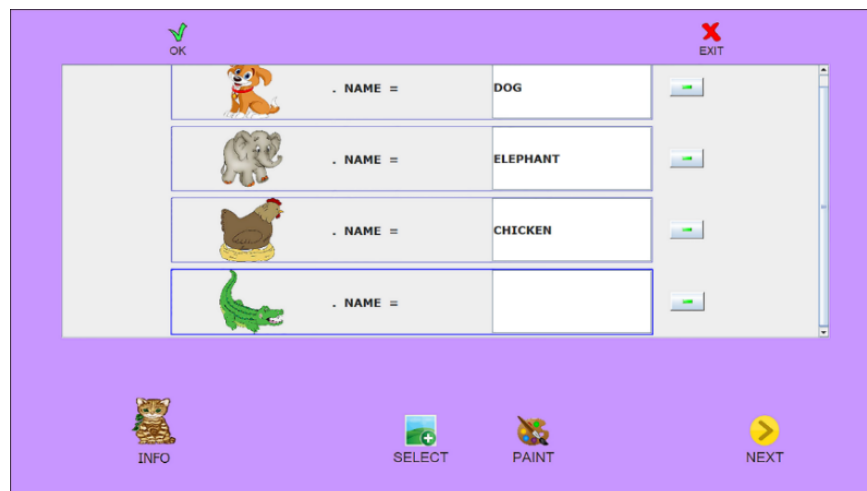
### 3.1. PhysGramming

We paid special attention to implementing a digital platform developmentally appropriate for four to eight-year-old children, the strong majority of which are novice users of digital technology [19,72]. Thence, we provide a digital tool to the educational community that: (a) maintains a "forgiving" character, by not showing emphasis on the mistakes the user makes and rewarding the successful attempts with a firework animation, accompanied with cheerful sounds [73], (b) does not require effort for finding tools, since it is kept uncomplicated [74] and (c) includes a brief tutorial [74]. It also contains bright colors [75,76] simple and big icons [73] and limited text of big font size [74,77]. Finally, its functionality is self-explained, based on a few components of increased size [74].

PhysGramming is compatible with common operating systems (Windows, Ubuntu, Android, etc.) and runs not only on personal computers but on smart mobile devices too, exploiting the advantages of adopting the use of smart mobile devices in compulsory education [72,78]. Actually, in a relevant research study of ours, harmonized with the shift of education to mobile technologies, we employed PhysGramming in a mobile learning context to investigate the association between abstraction skills and educational performance in the environmental study course in early primary school [72].

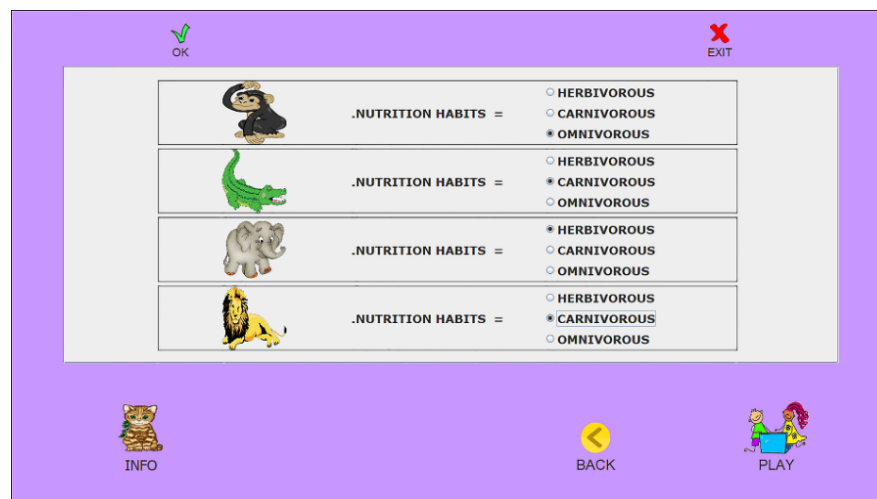
In order to better describe the functionality of PhysGramming and penetrate the philosophy of the proposed assessment tool, we will employ the thematic unit of the environmental study course, within the context of which this study took place i.e., animals' eating habits.

Since students are informed about the topic of the lesson, they have to decide which entities they are going to study. Then, they may paint, photograph or just select their images from PhysGramming’s image pool. All these entities are presented in command lines, in which students have to identify the entities (Figure 1).



**Figure 1.** Assigning values to the animals’ attribute “NAME”.

The next step is to identify the value of the attribute under investigation, which, in our case, is the selected animals’ eating habits (Figure 2). After identifying the values of the attributes “name” and “eating habits” for each one of the entities under investigation, PhysGramming has all the information needed to automatically construct digital games, which are unique just like students’ paintings and photos. In this article, we will discuss only the puzzle games PhysGramming creates.

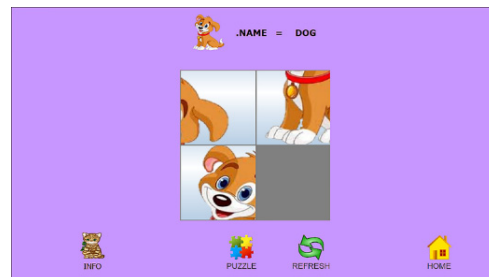


**Figure 2.** Assigning values to the animals’ attribute “NUTRITION HABITS”.

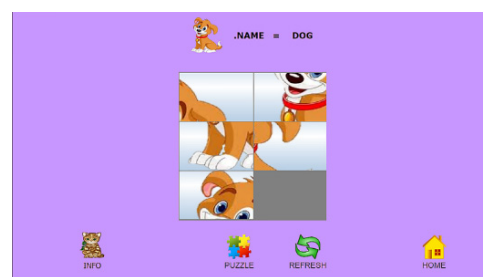
Puzzles do not require the information about the selected animals’ eating habits to be constructed. However, this information is a prerequisite for the construction of match-up and group games, which PhysGramming creates as part of a set of digital games, together with puzzles. Thus, although there is not an obvious correlation between puzzles and the environmental study thematic unit employed, they do relate to each other since puzzles are part of the set of digital games PhysGramming constructs.

PhysGramming constructs puzzles of four, six, nine, or twelve pieces (Figures 3–6). Each puzzle is a grid of randomly positioned pieces of an image, with the lower right cell

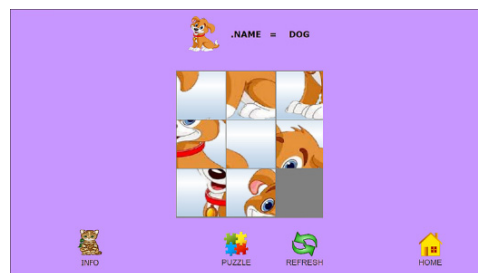
to be empty at first. In order to facilitate students' efforts we provide a guide picture placed above each puzzle.



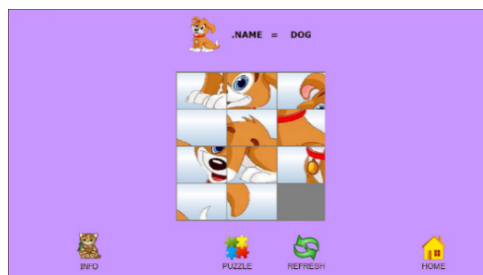
**Figure 3.** Four-piece dog puzzle.



**Figure 4.** Six-piece dog puzzle.



**Figure 5.** Nine-piece dog puzzle.



**Figure 6.** Twelve-piece dog puzzle.

The difficulty of solving the puzzles is analogous to the number of their pieces. Students have to reconstruct the puzzle's image by rearranging its pieces. Nevertheless, not all the pieces can be moved. In fact, only the pieces that are adjacent to the empty cell can be moved horizontally, vertically, or diagonally [19]. Therefore, solving a four-piece puzzle (Figure 3) is quite easy, since there is no restriction on moving its pieces. On the contrary, in all other cases (Figures 4–6), the pieces that can be moved are the ones adjacent to the empty cell. All the other pieces cannot be moved, until one of their adjacent cells becomes empty. In fact, the more pieces that make up the puzzle, the more the pieces cannot be moved until an adjoining cell empties, increasing the difficulty of solving.



Randomly repositioning the puzzle pieces will not facilitate students' solving attempts, except maybe in the case of the four-piece puzzles. In any other cases, solving a puzzle demands a solution plan. The levels of algorithmic thinking skills are determined by the number of pieces of the most difficult puzzle students manage to solve, assuming that the more the pieces of the puzzle, the more difficult it becomes to solve them, requiring higher algorithmic thinking levels (Table 1).

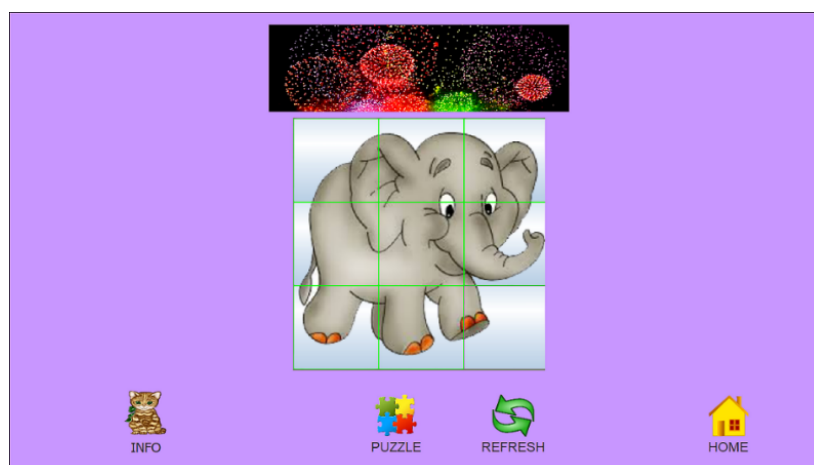
**Table 1.** Algorithmic thinking levels.

Algorithmic Thinking Levels	The Most Difficult Puzzle Solved
basic	four-piece puzzle
medium	six-piece puzzle
satisfactory	nine-piece puzzle
excellent	twelve-piece puzzle

The proposed algorithmic thinking levels are based on the grading scale employed in Greek primary schools, from third to sixth grade level i.e., Approximately good, Good, Very good, Excellent. Although our research efforts focus on first and second grade levels, we borrow this grading scale, because for the first and second grade students only descriptive evaluation is employed. The proposed differentiation of algorithmic thinking levels is in accordance with relevant research approaches documented in literature, such as, the work of Chongo et al. [79] that also proposes four CT skills levels i.e., Fail, Pass, Credit, Excellence.

To rank a student in a suggested algorithmic thinking level, it is enough to solve one puzzle of this level, which would be the most difficult puzzle they solved. For example, if we rank a student in satisfactory algorithmic thinking level, this means that they solved at least one nine-piece puzzle and did not solve a twelve-piece puzzle.

When a puzzle is solved a fireworks animation appears on the screen (Figure 7), accompanied by an audio reward. The use of audiovisual tools enhances the attractiveness and friendliness of PhysGramming and gives a boost to students to reach the established learning objectives [80]. As far as the goal of employing immediate reward is concerned, we seek to motivate students to intensify their efforts towards the learning objectives [81] and support them to remain focused on the learning activity they are occupied with [82,83]. Immediate reward also empowers friendly competition among students [84], advances their self-confidence [85] and amplifies their classmates' acceptance [83].

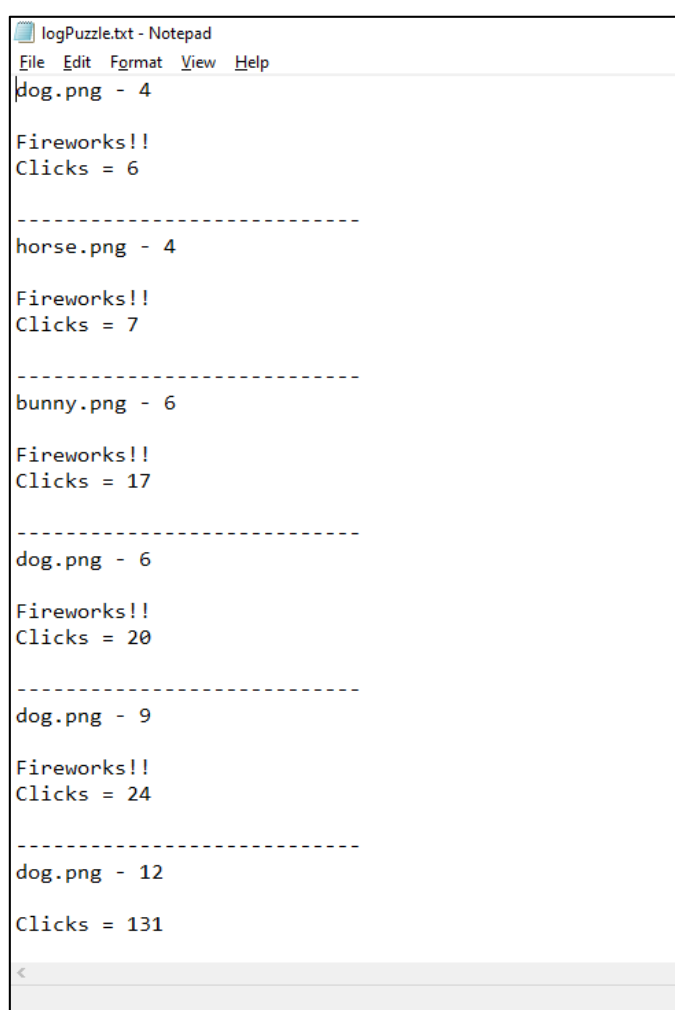


**Figure 7.** Rewarding students with fireworks animation.

PhysGramming is programmed to construct log files that keep information about all the attempts students make towards solving the puzzles. More specifically, log files provide information about each one of the puzzles solved, namely, the image of the puzzle, the

number of its pieces and the number of the moves students made to solve them. Relevant information is recorded about the students' failed attempts.

For example, let's consider the case of the log file presented in Figure 8, which provides evidence of the attempts a particular student made to solve puzzles. Dotted lines separate blocks of data. Each block contains data regarding the picture of each puzzle the students tried to solve, the number of pieces of each puzzle, and whether each attempt was successful or not. For example, the fourth data block reveals that the student tried and managed to solve a six-piece dog puzzle, having repositioned its pieces 20 times. The fact that "Fireworks!!" is recorded in the block means that fireworks appeared on the screen when the student solved the puzzle. On the contrary, the last data block informs us that the students tried hard to solve the twelve-piece dog puzzle, having repositioned its pieces 131 times. However, they finally failed to solve this particular puzzle, since "Fireworks!!" is not included in the data block.



```
logPuzzle.txt - Notepad
File Edit Format View Help
dog.png - 4

Fireworks!!
Clicks = 6

-----
horse.png - 4

Fireworks!!
Clicks = 7

-----
bunny.png - 6

Fireworks!!
Clicks = 17

-----
dog.png - 6

Fireworks!!
Clicks = 20

-----
dog.png - 9

Fireworks!!
Clicks = 24

-----
dog.png - 12

Clicks = 131

<
```

**Figure 8.** Log file.

### 3.2. Research Sample

Targeting at formulating a representative sample we employed the cluster sampling method [23]. The schools involved in the research study were from different areas of Heraklion city, in order to be sure that the students that would participate in the research study would come from several socio-economic strata. In this way, the final sample would be attributed with a wide range of learning achievements and attitudes towards the educational process [86,87]. As far as the sample size is concerned, we ended up with 435 first and second graders. The sample was gender-balanced—210 girls (48.28%) and

225 boys (51.72%). It was also grade-balanced—218 first graders (50.11%) and 217 s graders (49.89%).

### 3.3. Validation

In order to establish the validity and the reliability of the results provided by the proposed assessment tool, we took the adequate steps proposed in the international literature [23]. For some types of validity and reliability, repeated measurements were required, which were implemented on a sample of 75 primary school students of first and second grade. Although the objective of this article is not to present the relevant methods employed, we indicatively mention that Pearson correlation coefficient (Pearson's  $r$ ) is calculated to be 0.81. Since Pearson's  $r$  quantifies the strength of the relationship between test/retest scores [88], the calculated value indicates very good test/retest reliability i.e., very good stability of measurements obtained from the same individuals at different times [88,89].

## 4. Results

We set the hypothesis that algorithmic thinking levels are not associated to students' age and test it at a 5% level of significance i.e., with the predetermined alpha level of significance to be 0.05.

### 4.1. Examining the Hypothesis Set

We formulate the contingency table of observed frequencies (Table 2) and, based on its content, we construct the contingency table of expected frequencies. Then, we calculate chi-square to be 8.7543,  $df$  to be 3, and  $p$ -value to be 0.03274. Since  $p$ -value is less than 0.05, we reject the hypothesis that there is no correlation between algorithmic thinking levels and students' age. Hence, we admit that, as far as the first and second-graders are concerned, algorithmic thinking levels are related to age.

**Table 2.** Contingency table of observed frequencies.

Grade Level \ Algorithmic Thinking	First	Second	Sum
Excellent	39	43	82
Satisfactory	59	83	142
Medium	96	69	165
Basic	24	22	46
Sum	218	217	435

### 4.2. Odds Ratio

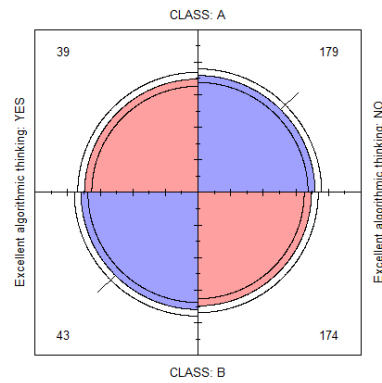
Next, we calculate the odds ratio for each algorithmic thinking level in relation to students' grade.

Based on the data presented in Table 2, for excellent algorithmic thinking, the odds ratio is calculated to be  $(39/179)/(43/174) = 0.218/0.247 = 0.883$ . Results indicate the odds ratio is less than one, and, thus, the probability of detecting excellent algorithmic thinking levels is higher for second graders. For satisfactory algorithmic thinking levels, the odds ratio is calculated to be 0.512 i.e., less than one, and, thus, the probability of detecting satisfactory algorithmic thinking levels is higher for second graders. On the contrary, for the cases of medium and basic algorithmic thinking levels the odds ratio is calculated to be greater than one (1.689 and 1.097, respectively), meaning that the probability of detecting these algorithmic thinking levels is higher for first graders.

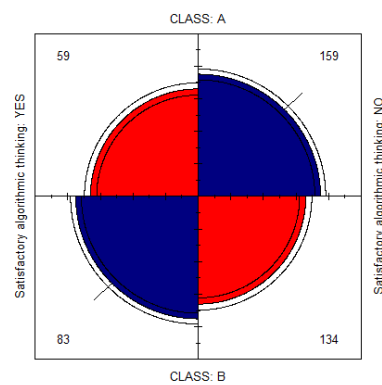
### 4.3. Data Visualization

We construct fourfold plots in order to graphically represent the correlation between algorithmic thinking levels and students' age [90]. Studying the displays (Figures 9–12), we conclude that the odds of identifying excellent or satisfactory algorithmic thinking skills is greater for second graders (Class B) than for first graders (Class A). On the contrary,

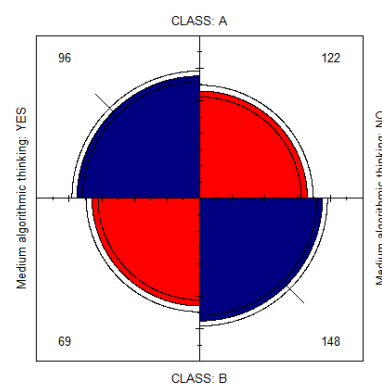
first graders are more likely to demonstrate medium or basic algorithmic thinking skills. Especially in the case of satisfactory and medium algorithmic thinking, the confidence rings are far away from overlapping and visualized in brighter colors, showing that these associations are highly significant [90]. In other words, medium algorithmic thinking is highly associated with first graders, while satisfactory algorithmic thinking is highly associated with second graders.



**Figure 9.** Fourfold display for excellent algorithmic thinking.



**Figure 10.** Fourfold display for satisfactory algorithmic thinking.



**Figure 11.** Fourfold display for medium algorithmic thinking.

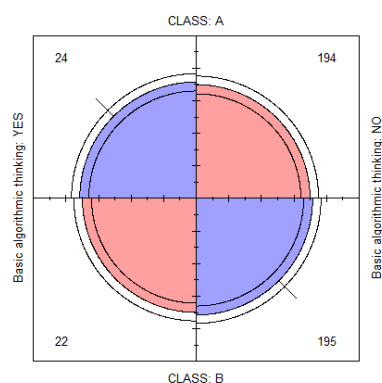


Figure 12. Fourfold display for basic algorithmic thinking.

Alternatively, we provide a sieve diagram (or parquet diagram), which plots rectangles with areas proportional to the expected frequencies and is filled with a number of squares corresponding to the observed frequencies. Hence, the densities visualize the deviations of the observed from the expected values [91]. Different colors are employed to point out whether the deviation from independence is positive or negative. In our case, the areas filled with blue squares signify the cases most likely to occur (Figure 13), indicating that medium and basic algorithmic thinking is more likely to characterize first graders (A Class), in contrast with satisfactory and excellent algorithmic thinking, which is more likely to be observed in second grade (B Class).

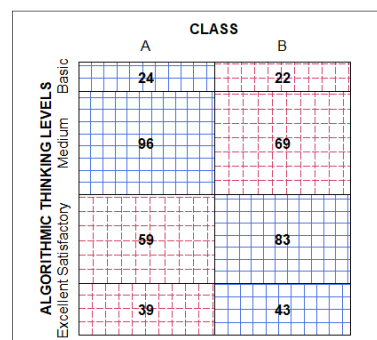


Figure 13. Sieve plot.

#### 4.4. Ordinal Logistic Regression Analysis

Next, we apply the ordinal logistic regression model considering algorithmic thinking levels as the dependent variable and students’ grade as the independent variable. Employing the polr function in the RStudio programming environment, we get the regression analysis results, which include the value of each coefficient, standard errors, and t-value (Table 3).

Table 3. Regression analysis results.

Coefficients:			
	Value	Std. Error	t-value
Grade Level	−0.2981	0.1756	−1.698
Intercepts:			
	Value	Std. Error	t-value
basic   excellent	−2.2954	0.1836	−12.5052
excellent   satisfactory	−1.0354	0.1425	−7.2649
satisfactory   medium	0.3402	0.1335	2.5480
Residual Deviance: 1115.307			
AIC: 1123.307			

The results indicate that the chances (in logarithmic form) second graders have to demonstrate basic algorithmic thinking levels (instead of medium, satisfactory or excellent) are 0.3 points lower than the chances first graders have.

We conclude to analogous results applying the exponential function. Indeed,  $\exp(-0.2981) = 0.742$ , meaning that for second graders, the chances of having excellent algorithmic thinking are 1.35 times ( $1/0.742 = 1.35$ ) higher than the first graders.

These results are verified through the calculation of predicted probabilities, employing the predict function in R (Table 4).

**Table 4.** Predicted probabilities.

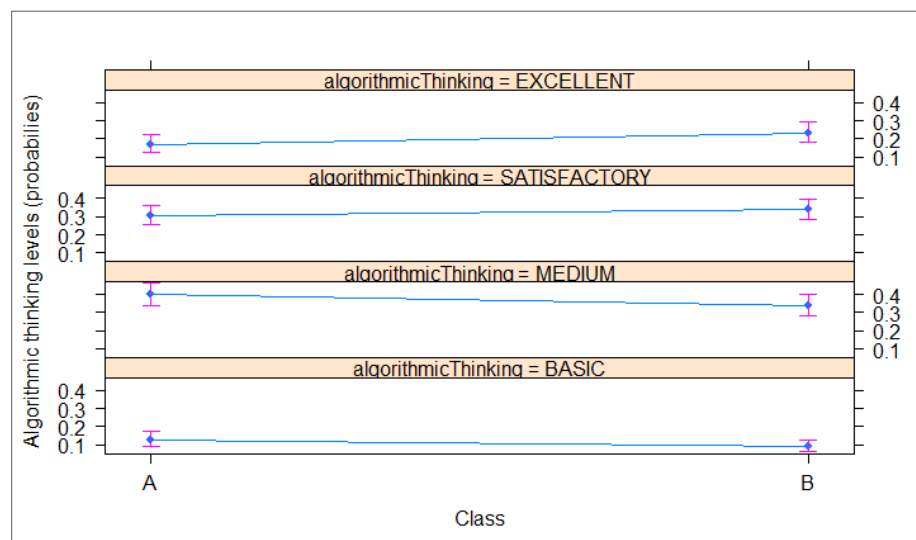
	Excellent	Satisfactory	Medium	Basic
First Grade	0.171	0.322	0.416	0.092
Second Grade	0.204	0.331	0.346	0.119

If we calculate the proportional odds for basic algorithmic thinking skills, we come in agreement with the results obtained so far. Indeed:

$$(0.092/(1 - 0.092))/(0.119/(1 - 0.119)) = (0.092/0.908)/(0.119/0.881) = 0.101/0.135 = 0.748$$

In the same way, we could calculate the proportional odds ratio for all the algorithmic thinking levels.

Finally, we employ the machine learning method [26], aiming at predicting the probability of new data to be placed at each one of the algorithmic thinking levels, in relation to students' grade. The dependent variable is the level of algorithmic thinking, while the independent variable is the grade (Figure 14).



**Figure 14.** Machine learning.

In all, 80% of the survey data was used to create the prediction equations and the remaining 20% to test them. The relevant graph shows that attending the second grade increases the probability of being classified at the higher levels of algorithmic thinking (satisfactory and excellent), while decreasing the probability of being classified at the lower levels of algorithmic thinking (basic and medium).

## 5. Discussion

The findings of our research study confirm the hypothesis set that algorithmic thinking skills are correlated with age in the first two years of primary education. Moving one step

ahead, our study results not only answer the research question positively but also reveal that age constitutes a predictor factor for algorithmic thinking levels in early childhood. Now, let us elaborate on how the research findings are confirmed by the existing relevant literature, and the contribution entailed, underlining the need of our study.

Several research studies point out that CT skills could be depicted as a reflection of skills related to cognitive development, including abstraction, algorithmic thinking, problem-solving, and critical thinking skills, reinforcing the opinion of a potential correlation between CT skill levels and educational grade levels [16,92].

A relevant study conducted in Spain on a sample of 1251 fifth and sixth graders confirms the correlation between the students' age and the levels of basic CT skills [93]. The same study confirms the hypothesis that CT skills are inextricably linked to problem-solving skills. In addition, it is emphasized that the level of cognitive development and the level of maturity are important factors in the development of CT skills and problem-solving ability [93].

A survey conducted in Ankara in 2015–2016 examined how specific parameters affect the development of core CT skills, such as algorithmic thinking. The research sample consisted of 156 students ranging from fifth to twelfth grade. Research revealed that the higher the grade, the higher the students' CT skills [16].

A research study conducted in the Netherlands investigated, among others, how age is related to students' success in CT tasks. The research sample consisted of 200 students six to twelve years old. The CT tasks pertained to abstraction and decomposition skills, which are inextricably related to algorithmic thinking skills. According to the research evidence, age seems to have a positive effect on the development of students' CT skills [17].

Another research study employed the framework CT-cube, which is appropriate for designing, realizing, analyzing and assessing CT activities, to design an unplugged task, called Cross Array Task, in order to assess the algorithmic thinking skills of K-12 students. The relevant study considered a sample of 109 students aged three to sixteen in Switzerland and concluded that algorithmic thinking skills of K-12 students increase with the age. However, this increase is observed mainly between lower and upper primary school students [94].

Assaying the results of our work in comparison with the literature, we notice that the findings of the aforementioned research studies are in line with those of our study. In fact, the last two surveys are particularly important for certifying our results, since part of the age group of their samples concerns children of the age group we are studying. Elaborating on the implications of our study, we should not be confined to its feature of being substantiated by the existing research background. In fact, we add to the efforts of the relevant research field by investigating, for the first time, the narrow range of the first two years of primary school. Until now, all the relevant studies examined the influence of age on the development of CT skills by employing samples of wider age ranges. The fact that the development of cognitive skills reflects the development of CT skills [16,92] does not imply that this development is observable in a narrow age range. In other words, we do know that the development of CT skills could be observed when we investigate wide age ranges. However, it is not obvious that this development is observable in a narrow age range, especially when it concerns the beginning of schooling.

Attempting to unpack how jigsaw puzzles exercise algorithmic thinking, we turn to works documented in the relevant literature. Shortly discussing some of these works, we support our decision to employ jigsaw puzzles in order to assess algorithmic thinking.

One of these studies proposes a project developmentally appropriate for students' age from eight to eleven, which aims to determine, examine and refine a learning path focused on stimulating CT competencies. Students are involved in collaborative game-making activities implemented on the Scratch platform. A type of these activities concerns checking the position and rotation of the pieces. More precisely, in the first activity, students have to drag the jigsaw puzzle pieces to their correct location, and in the second they have to appropriately rotate the pieces [95].

Another study examines the potential beneficial effect of including unplugged activities in the instructional settings in order to foster CT competencies in the early years of primary education. Digital jigsaw puzzle and unplugged puzzles are employed in the context of a quasi-experimental study, exploring and confirming the subsequent advantages of a mixed approach of both unplugged and plugged-in activities [69].

A hybrid schema of offline and online activities, which aims to develop essential CT skills and is developmentally appropriate for children four to six years old, is also documented. Among the proposed activities is breaking a big jigsaw puzzle into smaller pieces and then reconstructing it [96].

The studies discussed so far certify that jigsaw puzzles have already been employed successfully in order to support the development and the assessment of core CT competencies, such as algorithmic thinking, in compulsory education. Since our research interest focuses on young children, we indicatively presented studies conducted in primary education with the emphasis put on the early years.

Focusing on the sensitive years of early education, we provide a solid theoretical basis that could be useful to scholars engaged in investigating the introduction of CT skills at the beginning of schooling. Educators, researchers and policymakers could support the need disclosed by our study that the developmentally appropriate cultivation of CT skills in the first two years of primary school deserves the design and implementation of educational tools of ranked levels of difficulty, depending on students' age.

### 5.1. *The Rational of the Study*

As far as the rational of our research study is concerned, it is not limited to revealing the need for providing age-based activities in early childhood in order to promote CT skills, but also adds to the efforts of exploiting digital technology to support educational assessment. Having in mind how appealing digital environments can become to children [97] and building on the positive behaviors being developed in early educational settings that employ coding as a playground [41], we propose an assessment tool of playful and constructivist nature that provides an initial glance at the main principles of object-oriented programming, although no direct reference is made to them [19].

The assessment environment we propose supports the integration of CT in early STEM education, facilitating students to study science in authentic and computationally sophisticated manners [46]. The festive atmosphere established in the classroom and the creative competition developed among the students while applying the proposed assessment tool facilitated its embracement by the educational community that participated in the research study i.e., both students and teachers [72].

Finally, yet importantly, the proposed assessment tool adds to the efforts of convergent measurements of CT skills and learning achievements in STEM fields, addressing the CT assessment challenge by exploiting STEM educational scenarios [32]. In our study, we focused on the thematic unit of animals' eating habits, which is included in the textbook of the environmental study course in both first and second grade in Greece. Someone could argue that this topic appertains to the science of biology and could not formulate the basis of an environmental science scenario. In order to address this skepticism, we will shortly discuss the case of Bovine Spongiform Encephalopathy (BSE), commonly known as mad-cow disease. BSE is a paradigm that consolidates the significance of respecting animals' eating habits, since their disruption could provoke multifarious consequences affecting not only animals, but human beings too. In the case of BSE, herbivorous mammals were unnaturally turned into carnivorous, not to say cannibals, since they consumed feed that contained contaminated meat and bone meal as a protein meat source [98,99]. BSE firstly appeared as a veterinary problem in Great Britain, then it was upgraded to a public health hazard and, finally, it was metamorphosed into an international crisis of economic and political dimensions, affecting even the relationships of the European Union and the United States [98]. The fact that the prion proteins—the cause of BSE—persist in the soil entailed unbounded environmental hazard, putting human health at unquantifiable



risk [98,99]. In this sense, we endorse the study of animals' eating habits in early formal STEM settings, since it opens up horizons for fostering environmental ethics and provides a creative context for consolidating moral, social and political principles and values.

### 5.2. Limitations and Perspectives

Our future direction is to extend our research throughout the country, addressing the sample limitation of the present study, which we conducted in the city of Heraklion. We also plan to strengthen the outcomes of this study by conducting research in which we will employ several other thematic units of the environmental study course.

We also intend to provide evidence of how first and second graders strategize when solving the puzzles, elaborating on the way they approach them.

## 6. Conclusions

The learning science community recognizes that K-12 STEM fields have to follow the evolution of the 21st digital era and become more authentic by embracing CT [46]. Adding to the limited research efforts on developing and accessing CT in early learning settings [33,46], we examined the algorithmic thinking levels from an aged-based point of view. Our study was implemented within the context of the environmental study course, exploiting young children's engagement with science experiences [100]. Intrigued by the fact that, thus far, the relevant studies have investigated the effect of age on CT development in wide age ranges, we concentrated on corroborating the existing research findings, in the narrow age range of first and second graders. The study results confirm the correlation between algorithmic thinking skills and age, indicating that age constitutes a predictor factor for algorithmic thinking levels in early childhood.

This study is of interest to those who believe that teaching young children to code can accelerate the acquisition of CT skills [101], such as algorithmic thinking. To this end, we developed the digital platform PhysGramming that introduces a novel schema of text-based and visual programming techniques, providing young students their first contact with fundamental object-based programming concepts. Building on the efficacy of game-based learning in developing students' 21st-century skills, such as CT [102], and the effectiveness of digital games in producing environmental friendly attitudes and behaviors [103], we go beyond the narrow limits of the simple use of digital games and propose a game developing technique with STEM dimensions.

This study also concerns those who appreciate the value of multidisciplinary educational schemas, stand for the necessity of high-quality STEM education, and emphasize the need for bringing developmentally appropriate CT practices into STEM learning environments.

**Author Contributions:** Conceptualization, K.K. and M.K.; methodology, K.K. and M.K.; software, K.K.; validation, K.K. and M.K.; formal analysis, K.K. and M.K.; investigation, K.K.; resources, K.K. and M.K.; data curation, K.K. and M.K.; writing—original draft preparation, K.K.; writing—review and editing, K.K. and M.K.; visualization, K.K.; supervision, M.K.; project administration, M.K.; funding acquisition, K.K. and M.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** The study was conducted in accordance with the Declaration of Helsinki, and approved by the Institutional Review Board of the University of Crete (555/17 October 2018).

**Informed Consent Statement:** Written informed consent has been obtained from the participants to publish this paper.

**Data Availability Statement:** Publicly available datasets were analyzed in this study. This data can be found at <http://physgramming.edc.uoc.gr/main.html>.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Nordby, S.K.; Bjerke, A.H.; Mifsud, L. Computational thinking in the primary mathematics classroom: A systematic review. *Digit. Exp. Math. Educ.* **2022**, *8*, 27–49. [\[CrossRef\]](#)
2. Wing, J. Research notebook: Computational thinking—What and why. *Link Mag.* **2011**, *6*, 20–23.
3. Zhang, L.; Nouri, J. A systematic review of learning computational thinking through Scratch in K-9. *Comput. Educ.* **2019**, *141*, 103607. [\[CrossRef\]](#)
4. Wing, J.M. Computational thinking. *Commun. ACM* **2006**, *49*, 33–35. [\[CrossRef\]](#)
5. Acevedo-Borrega, J.; Valverde-Berrocoso, J.; Garrido-Arroyo, M.d.C. Computational Thinking and Educational Technology: A Scoping Review of the Literature. *Educ. Sci.* **2022**, *12*, 39. [\[CrossRef\]](#)
6. Shute, V.J.; Sun, C.; Asbell-Clarke, J. Demystifying computational thinking. *Educ. Res. Rev.* **2017**, *22*, 142–158. [\[CrossRef\]](#)
7. Yang, D.; Baek, Y.; Ching, Y.H.; Swanson, S.; Chittoori, B.; Wang, S. Infusing Computational Thinking in an Integrated STEM Curriculum: User Reactions and Lessons Learned. *EJSTEME Eur. J. STEM Educ.* **2021**, *6*, 4. [\[CrossRef\]](#)
8. Grover, S.; Fisler, K.; Lee, I.; Yadav, A. Integrating Computing and Computational Thinking into K-12 STEM Learning. In Proceedings of the 51st ACM Technical Symposium on Computer Science Education, Portland, OR, USA, 11–14 March 2020; pp. 481–482. [\[CrossRef\]](#)
9. Waterman, K.P.; Goldsmith, L.; Pasquale, M. Integrating computational thinking into elementary science curriculum: An examination of activities that support students' computational thinking in the service of disciplinary learning. *J. Sci. Educ. Technol.* **2020**, *29*, 53–64. [\[CrossRef\]](#)
10. Hutchins, N.M.; Biswas, G.; Maróti, M.; Lédeczi, Á.; Grover, S.; Wolf, R.; Blair, K.P.; Chin, D.; Conlin, L.; Basu, S.; et al. C2STEM: A system for synergistic learning of physics and computational thinking. *J. Sci. Educ. Technol.* **2020**, *29*, 83–100. [\[CrossRef\]](#)
11. Sung, W.; Ahn, J.; Black, J.B. Introducing computational thinking to young learners: Practicing computational perspectives through embodiment in mathematics education. *Technol. Knowl. Learn.* **2017**, *22*, 443–463. [\[CrossRef\]](#)
12. Ardoin, N.M.; Bowers, A.W. Early childhood environmental education: A systematic review of the research literature. *Educ. Res. Rev.* **2020**, *31*, 100353. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Malyn-Smith, J.; Lee, I.A.; Martin, F.; Grover, S.; Evans, M.A.; Pillai, S. Developing a framework for computational thinking from a disciplinary perspective. In Proceedings of the International Conference on Computational Thinking Education, Hong Kong, China, 14–16 June 2018; pp. 182–186.
14. Swaid, S.I. Bringing computational thinking to STEM education. *Procedia Manuf.* **2015**, *3*, 3657–3662. [\[CrossRef\]](#)
15. Hsu, T.C.; Chang, S.C.; Hung, Y.T. How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Comput. Educ.* **2018**, *126*, 296–310. [\[CrossRef\]](#)
16. Durak, H.Y.; Saritepeci, M. Analysis of the relation between computational thinking skills and various variables with the structural equation model. *Comput. Educ.* **2018**, *116*, 191–202. [\[CrossRef\]](#)
17. Rijke, W.J.; Bollen, L.; Eysink, T.H.; Tolboom, J.L. Computational Thinking in Primary School: An Examination of Abstraction and Decomposition in Different Age Groups. *Inform. Educ.* **2018**, *17*, 77–92. [\[CrossRef\]](#)
18. Jiang, B.; Li, Z. Effect of Scratch on computational thinking skills of Chinese primary school students. *J. Comput. Educ.* **2021**, *8*, 505–525. [\[CrossRef\]](#)
19. Kanaki, K.; Kalogiannakis, M. Introducing fundamental object-oriented programming concepts in preschool education within the context of physical science courses. *Educ. Inf. Technol.* **2018**, *23*, 2673–2698. [\[CrossRef\]](#)
20. Breien, F.S.; Wasson, B. Narrative categorization in digital game-based learning: Engagement, motivation & learning. *Br. J. Educ. Technol.* **2021**, *52*, 91–111. [\[CrossRef\]](#)
21. Rushton, S.; Juola-Rushton, A.; Larkin, E. Neuroscience, play and early childhood education: Connections, implications and assessment. *Early Child. Educ. J.* **2010**, *37*, 351–361. [\[CrossRef\]](#)
22. Sigman, M.; Peña, M.; Goldin, A.P.; Ribeiro, S. Neuroscience and education: Prime time to build the bridge. *Nat. Neurosci.* **2014**, *17*, 497–502. [\[CrossRef\]](#)
23. Cohen, L.; Manion, L.; Morrison, K. *Research Methods in Education*, 3rd ed.; Routledge: London, UK, 2007.
24. Petousi, V.; Sifaki, E. Contextualizing harm in the framework of research misconduct. Findings from discourse analysis of scientific publications. *Int. J. Sustain. Dev.* **2020**, *23*, 149–174. [\[CrossRef\]](#)
25. Kanaki, K.; Kalogiannakis, M. Assessing algorithmic thinking skills in relation to gender in early childhood. *Educ. Process Int. J.* **2022**; *in press*.
26. Nafea, I.T. Machine Learning in Educational Technology. In *Machine Learning-Advanced Techniques and Emerging Applications*; Farhadi, H., Ed.; IntechOpen: London, UK, 2008; pp. 175–183. [\[CrossRef\]](#)
27. Li, Y.; Schoenfeld, A.H.; di Sessa, A.A.; Graesser, A.C.; Benson, L.C.; English, L.D.; Duschl, R.A. On computational thinking and STEM education. *J. STEM Educ. Res.* **2020**, *3*, 147–166. [\[CrossRef\]](#)
28. Grizioti, M.; Kynigos, C. Children as players, modders, and creators of simulation games: A design for making sense of complex real-world problems. In Proceedings of the 20th ACM Conference on Interaction Design and Children, London, UK, 21–24 June 2020; pp. 363–374. [\[CrossRef\]](#)
29. Grover, S. Assessing Algorithmic and Computational Thinking in K-12: Lessons from a Middle School Classroom. In *Emerging Research, Practice, and Policy on Computational Thinking*; Rich, P., Hodges, C., Eds.; Educational Communications and Technology: Issues and Innovations; Springer International Publishing AG: Berlin/Heidelberg, Germany, 2017; pp. 269–288. [\[CrossRef\]](#)

30. Poulakis, E.; Politis, P. Computational Thinking Assessment: Literature Review. In *Research on E-Learning and ICT in Education*; Tsiatsos, T., Demetriadis, S., Mikropoulos, A., Dagdilelis, V., Eds.; Springer: Cham, Switzerland, 2021; pp. 111–128. [CrossRef]
31. Tang, H.; Xu, Y.; Lin, A.; Heidari, A.A.; Wang, M.; Chen, H.; Luo, Y.; Li, C. Predicting green consumption behaviors of students using efficient firefly grey wolf-assisted K-nearest neighbor classifiers. *IEEE Access* **2020**, *8*, 35546–35562. [CrossRef]
32. Román-González, M.; Moreno-León, J.; Robles, G. Combining assessment tools for a comprehensive evaluation of computational thinking interventions. In *Computational Thinking Education*; Kong, S.C., Abelson, H., Eds.; Springer: Singapore, 2019; pp. 79–98. [CrossRef]
33. Tsarava, K.; Moeller, K.; Román-González, M.; Golle, J.; Leifheit, L.; Butz, M.V.; Ninaus, M. A cognitive definition of computational thinking in primary education. *Comput. Educ.* **2022**, *179*, 104425. [CrossRef]
34. Schroth, S.T.; Daniels, J. *Building STEM Skills through Environmental Education*; IGI Global: Hershey, PA, USA, 2021. [CrossRef]
35. Levin, I.; Mamlok, D. Culture and society in the digital age. *Information* **2021**, *12*, 68. [CrossRef]
36. Frankenreiter, J.; Livermore, M.A. Computational methods in legal analysis. *Annu. Rev. Law Soc. Sci.* **2020**, *16*, 39–57. [CrossRef]
37. Kharchenko, P.V. The triumphs and limitations of computational methods for scRNA-seq. *Nat. Methods* **2021**, *18*, 723–732. [CrossRef]
38. Mejia, C.; D'Ippolito, B.; Kajikawa, Y. Major and recent trends in creativity research: An overview of the field with the aid of computational methods. *Creat. Innov. Manag.* **2021**, *30*, 475–497. [CrossRef]
39. Lodi, M.; Martini, S. Computational thinking, between Papert and Wing. *Sci. Educ.* **2021**, *30*, 883–908. [CrossRef]
40. Kanaki, K.; Kalogiannakis, M.; Stamovlasis, D. Assessing Algorithmic Thinking Skills in Early Childhood Education: Evaluation in Physical and Natural Science Courses. In *Handbook of Research on Tools for Teaching Computational Thinking in P-12 Education*; IGI Global: Hershey, PA, USA, 2020; pp. 104–139. [CrossRef]
41. Bers, M.U.; González-González, C.; Armas-Torres, M.B. Coding as a playground: Promoting positive learning experiences in childhood classrooms. *Comput. Educ.* **2019**, *138*, 130–145. [CrossRef]
42. Saqr, M.; Ng, K.; Oyelere, S.S.; Tedre, M. People, ideas, milestones: A scientometric study of computational thinking. *ACM Trans. Comput. Educ.* **2021**, *21*, 20. Available online: [https://www.researchgate.net/publication/347583738\\_People\\_Ideas\\_Milestones\\_A\\_Scientometric\\_Study\\_of\\_Computational\\_Thinking](https://www.researchgate.net/publication/347583738_People_Ideas_Milestones_A_Scientometric_Study_of_Computational_Thinking) (accessed on 27 May 2022). [CrossRef]
43. Sanford, J.F.; Naidu, J.T. Computational thinking concepts for grade school. *Contemp. Issues Educ. Res. CIER* **2016**, *9*, 23–32. [CrossRef]
44. Lye, S.Y.; Koh, J.H.L. Review on teaching and learning of computational thinking through programming: What is next for K-12? *Comput. Hum. Behav.* **2014**, *41*, 51–61. [CrossRef]
45. Wing, J.M. Computational thinking and thinking about computing. *Philos. Trans. R. Soc. A* **2008**, *366*, 3717–3725. [CrossRef] [PubMed]
46. Grover, S.; Biswas, G.; Dicks, A.C.; Farris, A.V.; Sengupta, P.; Covitt, B.A.; Gunckel, K.L.; Berkowitz, A.; Moore, J.C.; Irgens, G.A.; et al. Integrating STEM and computing in PK-12: OperationalizPeople, ideas, milestones: A scientometric study of computational thinking for STEM learning and assessment. In Proceedings of the Interdisciplinarity of the Learning Sciences, 14th International Conference of the Learning Sciences (ICLS), Nashville, TN, USA, 19–23 June 2020.
47. NGSS Lead States. *Next Generation Science Standards: For States, by States*; The National Academies Press: Washington, DC, USA, 2013. Available online: [https://epsc.wustl.edu/seismology/book/presentations/2014\\_Promotion/NGSS\\_2013.pdf](https://epsc.wustl.edu/seismology/book/presentations/2014_Promotion/NGSS_2013.pdf) (accessed on 12 April 2022).
48. Futschek, G. Algorithmic thinking: The key for understanding computer science. In Proceedings of the International Conference on Informatics in Secondary Schools-Evolution and Perspectives, Vilnius, Lithuania, 7–11 November 2006; pp. 159–168. [CrossRef]
49. Tengler, K.; Kastner-Hauler, O.; Sabitzer, B. Enhancing Computational Thinking Skills using Robots and Digital Storytelling. In Proceedings of the CSEDU, Online Conference, 23–25 April 2021; pp. 157–164. [CrossRef]
50. Figueiredo, M.P.; Amante, S.; Gomes, H.M.D.S.V.; Gomes, C.A.; Rego, B.; Alves, V.; Duarte, R.P. Algorithmic Thinking in Early Childhood Education: Opportunities and Supports in the Portuguese Context. In Proceedings of the EduLearn 2021, Online Conference, 5–6 July 2021; pp. 9339–9348. [CrossRef]
51. Vujčić, L.; Jančec, L.; Mezak, J. Development of algorithmic thinking skills in early and preschool education. In Proceedings of the EDULEARN21, Online Conference, 5–6 July 2021. [CrossRef]
52. Labusch, A.; Eickelmann, B.; Vennemann, M. Computational thinking processes and their congruence with problem-solving and information processing. In *Computational Thinking Education*; Kong, S.C., Abelson, H., Eds.; Springer: Singapore, 2019; pp. 65–78. [CrossRef]
53. Dagienė, V.; Futschek, G. Bebras international contest on informatics and computer literacy: Criteria for good tasks. In Proceedings of the International Conference on Informatics in Secondary Schools-Evolution and Perspectives, Torun, Poland, 1–4 July 2008; pp. 19–30. [CrossRef]
54. Burton, B.A. Encouraging Algorithmic Thinking Without a Computer. *Olymp. Inform.* **2010**, *4*, 3–14.
55. Merry, B.; Gallotta, M.; Hultquist, C. Challenges in running a computer olympiad in South Africa. *Olymp. Inform.* **2008**, *2*, 105–114.
56. Chen, G.; Shen, J.; Barth-Cohen, L.; Jiang, S.; Huang, X.; Eltoukhy, M. Assessing elementary students' computational thinking in everyday reasoning and robotics programming. *Comput. Educ.* **2017**, *109*, 162–175. [CrossRef]
57. Werner, L.; Denner, J.; Campe, S. Children programming games: A strategy for measuring computational learning. *ACM Trans. Comput. Educ.* **2015**, *14*, 24. [CrossRef]

58. Asbell-Clarke, J.; Rowe, E.; Almeda, V.; Edwards, T.; Bardar, E.; Gasca, S.; Baker, R.S.; Scruggs, R. The development of students' computational thinking practices in elementary-and middle-school classes using the learning game, Zoombinis. *Comput. Hum. Behav.* **2021**, *115*, 106587. [CrossRef]
59. Rowe, E.; Almeda, M.V.; Asbell-Clarke, J.; Scruggs, R.; Baker, R.; Bardar, E.; Gasca, S. Assessing implicit computational thinking in Zoombinis puzzle gameplay. *Comput. Hum. Behav.* **2021**, *120*, 106707. [CrossRef]
60. Kalogiannakis, M.; Papadakis, S.; Zourmpakis, A.I. Gamification in Science Education. A Systematic Review of the Literature. *Educ. Sci.* **2021**, *11*, 22. [CrossRef]
61. Misra, R.; Eyombo, L.; Phillips, F.T. Benefits and Challenges of Using Educational Games. In *Research Anthology on Developments in Gamification and Game-Based Learning*; IGI Global: Hershey, PA, USA, 2022; pp. 1560–1570. [CrossRef]
62. Kiss, G.; Arki, Z. The influence of game-based programming education on the algorithmic thinking. *Procedia Soc. Behav. Sci.* **2017**, *237*, 613–617. [CrossRef]
63. Gallagher, A.C. Jigsaw puzzles with pieces of unknown orientation. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 382–389.
64. Huroyan, V.; Lerman, G.; Wu, H.T. Solving jigsaw puzzles by the graph connection Laplacian. *SIAM J. Imaging Sci.* **2020**, *13*, 1717–1753. [CrossRef]
65. Doherty, M.J.; Wimmer, M.C.; Gollek, C.; Stone, C.; Robinson, E.J. Piecing together the puzzle of pictorial representation: How jigsaw puzzles index metacognitive development. *Child Dev.* **2021**, *92*, 205–221. [CrossRef]
66. Paikin, G.; Tal, A. Solving multiple square jigsaw puzzles with missing pieces. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4832–4839. [CrossRef]
67. Pomeranz, D.; Shemesh, M.; Ben-Shahar, O. A fully automated greedy square jigsaw puzzle solver. In Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA, 20–25 June 2011; pp. 9–16. [CrossRef]
68. Zhao, Y.X.; Su, M.C.; Chou, Z.L.; Lee, J. A puzzle solver and its application in speech descrambling. In Proceedings of the International Conference on Computer Engineering and Applications, Gold Coast, Australia, 17–19 January 2007; pp. 171–176.
69. del Olmo-Muñoz, J.; Cózar-Gutiérrez, R.; González-Calero, J.A. Computational thinking through unplugged activities in early years of Primary Education. *Comput. Educ.* **2020**, *150*, 103832. [CrossRef]
70. Ferrari, A.; Poggi, A.; Tomaiuolo, M. Object oriented puzzle programming. *Mondo. Digit.* **2016**, *15*, 64.
71. Janke, E.; Brune, P.; Wagner, S. Does outside-in teaching improve the learning of object-oriented programming? In Proceedings of the 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Florence, Italy, 16–24 May 2015; Volume 2, pp. 408–417. [CrossRef]
72. Kanaki, K.; Kalogiannakis, M.; Poulakis, E.; Politis, P. Employing Mobile Technologies to Investigate the Association Between Abstraction Skills and Performance in Environmental Studies in Early Primary School. *Int. J. Interact. Mob. Technol.* **2022**, *16*. [CrossRef]
73. Kraveva, R. Designing an Interface for a Mobile Application Based on Children's Opinion. *Int. J. Interact. Mob. Technol.* **2017**, *11*, 53–70. [CrossRef]
74. Nam, H. Designing User Experiences for Children. 2010. Available online: <https://www.uxmatters.com/mt/archives/2010/05/designing-user-experiences-for-children.php> (accessed on 13 March 2022).
75. Takahashi, F.; Kawabata, Y. The association between colors and emotions for emotional words and facial expressions. *Color Res. Appl.* **2018**, *43*, 247–257. [CrossRef]
76. Tham, D.S.Y.; Sowden, P.T.; Grandison, A.; Franklin, A.; Lee, A.K.W.; Ng, M.; Park, J.; Pang, W.; Zhao, J. A systematic investigation of conceptual color associations. *J. Exp. Psychol. Gen.* **2020**, *149*, 1311. [CrossRef] [PubMed]
77. Fruth, J.; Schulze, C.; Rohde, M.; Dittmann, J. E-learning of IT security threats: A game prototype for children. In Proceedings of the IFIP International Conference on Communications and Multimedia Security, Magdeburg, Germany, 25–26 September 2013; pp. 162–172. [CrossRef]
78. Criollo-C, S.; Guerrero-Arias, A.; Jaramillo-Alcázar, Á.; Luján-Mora, S. Mobile learning technologies for education: Benefits and pending issues. *Appl. Sci.* **2021**, *11*, 4111. [CrossRef]
79. Chongo, S.; Osman, K.; Nayan, N.A. Level of Computational Thinking Skills among Secondary Science Student: Variation across Gender and Mathematics Achievement. *Sci. Educ. Int.* **2020**, *31*, 159–163. [CrossRef]
80. McManis, L.D.; Gunnewig, S.B. Finding the education in educational technology with early learners. *Young Child* **2012**, *67*, 14.
81. Shoukry, L.; Sturm, C.; Galal-Edeen, G.H. Pre-MEGa: A proposed framework for the design and evaluation of preschoolers' mobile educational games. In *Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*; Elleithy, K., Sobh, T., Eds.; Springer: Cham, Switzerland, 2015; pp. 385–390.
82. Peijnenborgh, J.C.; Hurks, P.P.; Aldenkamp, A.P.; van der Spek, E.D.; Rauterberg, M.G.; Vles, J.S.; Hendriksen, J.G. A study on the validity of a computer-based game to assess cognitive processes, reward mechanisms, and time perception in children aged 4–8 years. *JMIR Serious Games* **2016**, *4*, e15. [CrossRef]
83. Richter, G.; Raban, D.R.; Rafaeli, S. Studying gamification: The effect of rewards and incentives on motivation. In *Gamification in Education and Business*; Reiners, T., Wood, L.C., Eds.; Springer: Cham, Switzerland, 2015; pp. 21–46. [CrossRef]
84. Montola, M.; Nummenmaa, T.; Lucero, A.; Boberg, M.; Korhonen, H. Applying game achievement systems to enhance user experience in a photo sharing service. In Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era, Tampere, Finland, 30 September–2 October 2009; pp. 94–97. [CrossRef]

85. Bleumers, L.; All, A.; Mariën, I.; Schurmans, D.; Van Looy, J.; Jacobs, A.; Willaert, K.; de Grove, F. *State of Play of Digital Games for Empowerment and Inclusion: A Review of the Literature and Empirical Cases*; EUR 25652; Publications Office of the European Union: Luxembourg, 2012; p. JRC77655. [[CrossRef](#)]
86. Bempechat, J.; Shernoff, D.J. Parental influences on achievement motivation and student engagement. In *Handbook of Research on Student Engagement*; Christenson, S., Reschly, A., Wylie, C., Eds.; Springer: Boston, MA, USA, 2012; pp. 315–342. [[CrossRef](#)]
87. Tan, C.Y.; Lyu, M.; Peng, B. Academic benefits from parental involvement are stratified by parental socioeconomic status: A meta-analysis. *Parent. Sci. Pract.* **2020**, *20*, 241–287. [[CrossRef](#)]
88. Vaz, S.; Falkmer, T.; Passmore, A.E.; Parsons, R.; Andreou, P. The case for using the repeatability coefficient when calculating test–retest reliability. *PLoS ONE* **2013**, *8*, e73990. [[CrossRef](#)]
89. Sullivan, G.M. A primer on the validity of assessment instruments. *J. Grad. Med. Educ.* **2011**, *3*, 119–120. [[CrossRef](#)]
90. Friendly, M. Visualizing categorical data: Data, stories, and pictures. In Proceedings of the Twenty-Fifth Annual SAS Users Group International Conference, Indiana Convention Center, Indianapolis, Indiana, 9–12 April 2000.
91. Meyer, D.; Zeileis, A.; Hornik, K. Visualizing contingency tables. In *Handbook of Data Visualization*; Springer Handbooks Comp., Statistics; Chen, C., Härdle, W., Unwin, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 589–616. [[CrossRef](#)]
92. Grover, S.; Pea, R. Computational thinking in K–12: A review of the state of the field. *Educ. Res.* **2013**, *42*, 38–43. [[CrossRef](#)]
93. Román-González, M.; Pérez-González, J.C.; Jiménez-Fernández, C. Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Comput. Hum. Behav.* **2017**, *72*, 678–691. [[CrossRef](#)]
94. Piatti, A.; Adorni, G.; El-Hamamsy, L.; Negrini, L.; Assaf, D.; Gambardella, L.; Mondada, F. The CT-cube: A framework for the design and the assessment of computational thinking activities. *Comput. Hum. Behav. Rep.* **2022**, *5*, 100166. [[CrossRef](#)]
95. Freina, L.; Bottino, R.; Ferlino, L. Fostering Computational Thinking skills in the Last Years of Primary School. *Int. J. Serious Games* **2019**, *6*, 101–115. [[CrossRef](#)]
96. Rompapas, D.; Steven, Y.; Chan, J. A Hybrid Approach to Teaching Computational Thinking at a K-1 and K-2 Level. In Proceedings of the CTE-STEM 2021: Fifth APSCE International Conference on Computational Thinking and STEM Education 2021, Virtual Conference, 2–4 June 2021; National Institute of Education, Nanyang Technological University: Singapore, 2021.
97. Zevenbergen, R.; Logan, H. Computer use by preschool children: Rethinking practice as digital natives come to preschool. *Australas. J. Early Child.* **2008**, *33*, 37–44. [[CrossRef](#)]
98. Freeman, H.B. Trade Epidemic: The Impact of the Mad Cow Crisis on EU-US Relations. *Boston Coll. Int. Comp. Law Rev.* **2002**, *25*, 343–371.
99. Washer, P. Representations of mad cow disease. *Soc. Sci. Med.* **2006**, *62*, 457–466. [[CrossRef](#)]
100. Fragkiadaki, G.; Ravanis, K. The unity between intellect, affect, and action in a child’s learning and development in science. *Learn. Cult. Soc. Interact.* **2021**, *29*, 100495. [[CrossRef](#)]
101. Relkin, E.; de Ruiter, L.E.; Bers, M.U. Learning to code and the acquisition of computational thinking by young children. *Comput. Educ.* **2021**, *169*, 104222. [[CrossRef](#)]
102. Qian, M.; Clark, K.R. Game-based Learning and 21st century skills: A review of recent research. *Comput. Hum. Behav.* **2016**, *63*, 50–58. [[CrossRef](#)]
103. Janakiraman, S.; Watson, S.L.; Watson, W.R.; Newby, T. Effectiveness of digital games in producing environmentally friendly attitudes and behaviors: A mixed methods study. *Comput. Educ.* **2021**, *160*, 104043. [[CrossRef](#)]