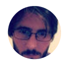
Development of Computational Thinking Skills through Unplugged Activities in Primary School

Conference Paper · November 2017

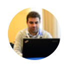
DOI: 10.1145/3137065.3137069

Christian Puhlmann Brackmann
Instituto Federal de Educação, Ciência e Tecnologia Farroupilha



Marcos Román-González
National Distance Education University

Gregorio Robles
King Juan Carlos University



Jesús Moreno-León
King Juan Carlos University

Development of Computational Thinking Skills through Unplugged Activities in Primary School

Christian P. Brackmann
Instituto Federal Farroupilha (IFFAR)
Santa Maria, Rio Grande do Sul, Brazil
brackmann@iffarroupilha.edu.br

Marcos Román-González
Universidad Nacional de Educación a Distancia (UNED)
Madrid, Spain
mroman@edu.uned.es

Gregorio Robles
Universidad Rey Juan Carlos (URJC)
Fuenlabrada, Madrid, Spain
grex@gsyc.urjc.es

Jesús Moreno-León
Universidad Rey Juan Carlos (URJC)
Fuenlabrada, Madrid, Spain
jesus.moreno@programamos.es

Ana Casali
Univ. Nacional de Rosario (UNR)
Rosario, Santa Fe, Argentina
acasali@fceia.unr.edu.ar

Dante Barone
U. Fed. do Rio Grande do Sul (UFRGS)
P. Alegre, Rio Grande do Sul, Brazil
barone@inf.ufrgs.br

ABSTRACT

Computational thinking is nowadays being widely adopted and investigated. Educators and researchers are using two main approaches to teach these skills in schools: with computer programming exercises, and with unplugged activities that do not require the use of digital devices or any kind of specific hardware. While the former is the mainstream approach, the latter is especially important for schools that do not have proper technology resources, Internet connections or even electrical power. However, there is a lack of investigations that prove the effectiveness of the unplugged activities in the development of computational thinking skills, particularly for primary schools. This paper, which summarizes a quasi-experiment carried out in two primary schools in Spain, tries to shed some light on this regard. The results show that students in the experimental groups, who took part in the unplugged activities, enhanced their computational thinking skills significantly more than their peers in the control groups who did not participate during the classes, proving that the unplugged approach may be effective for the development of this ability.

CCS CONCEPTS

• **Social and professional topics → Computational thinking**; **Computational science and engineering education**; **Computing literacy**;

KEYWORDS

Computational Thinking Unplugged, Evaluation, Computers in Education, Primary School, Elementary Education, Computational Thinking Test, Assessment

ACM Reference format:
Christian P. Brackmann, Marcos Román-González, Gregorio Robles, Jesús Moreno-León, Ana Casali, and Dante Barone. 2017. Development of Computational Thinking Skills through Unplugged Activities in Primary School. In *Proceedings of 12th Workshop in Primary and Secondary Computing Education, Nijmegen, Netherlands, November 8–10, 2017 (WiPSCE '17),* 8 pages.
https://doi.org/10.1145/3137065.3137069

DISCLAIMER

This document is a draft version. Final, published version can be accessed at ACM Digital Library

1 INTRODUCTION

In the last years, countries from all over the world have started to modify their national curricula to introduce Computational Thinking (CT) skills [4, 7]. A review of policy initiatives for integrating CT in compulsory education in European countries reveals two reasons behind this movement: i) to prepare for future employment and fill ICT job vacancies; and ii) to enable students to think in a different way, express themselves using new media and solve real-world problems [6].

Although the most common strategy to teach CT skills uses computerized activities mainly based on different types of programming tasks, educators and scholars are also using another approach with unplugged activities (i.e., in which there is no use of digital devices) [18]. Such activities involve logic games, cards, strings or physical movements that are used to represent and understand computer science concepts such as algorithms or data transmission.

The unplugged approach is the only one possible for a huge number of schools around the world that do not have basic technology infrastructure [33], such as electricity, Internet, computers, mobile devices, and other electronic devices. According to UNESCO, the use of ICT in education is still at a very early stage in most countries in sub-Saharan Africa, since the percentage of basic infrastructures in primary schools is under 15% in all the region [35]. In other regions, such as Asia, the percentage of schools with basic infrastructure is also far from being close to 100% [34]. But even in most European countries, there are still remote, rural areas with a lack of proper resources.

In this scenario, it is of capital importance to perform research that analyzes the effectiveness of the unplugged approach for the teaching of CT skills. This is the main goal of the investigation reported in this paper, in which we collaborated with two primary schools in Spain to perform a quasi-experiment to study differences

WiPSCE '17, November 8–10, 2017, Nijmegen, Netherlands

ACM ISBN 978-1-4503-5428-8/17/11…$15.00
https://doi.org/10.1145/3137065.3137069

in the development of CT skills between learners who participated in a series of unplugged activities, and students who did not take those lessons.

In addition, if evidences of the effectiveness of the unplugged approach are found, it would reinforce the theory that CT is mainly a problem-solving cognitive process/ability, which is possible to develop not only trough computer programming [36] [37].

The paper is structured as follows. In Section 2 we review research using the unplugged approach to teach computer science concepts and CT skills in schools. Then, in Section 3, we introduce the methods used during the intervention, including a description of the participants, instruments, class sessions, and other procedures. In Sections 4 and 5, we present and discuss the results and limitations, respectively. Finally, the main conclusions are summarized in Section 6, where we also discuss ideas for future research.

2 BACKGROUND

The first records of unplugged activities are found in 1997 when Bell published a draft version of "Computer Science Unplugged... Off-line activities and games for all ages", which was published in 1998 [5]. The book was targeted mainly for primary and secondary teachers, and it was very well accepted by educators and scholars alike. Due to the quality of the material, it was recommended by the Association for Computing Machinery (ACM) as part of the Computer Science Teachers Association school curriculum [3] and the activities were published on the CS Unplugged web page[1].

Although the use of computer programming activities is the main approach to teach CT skills in schools, educators and scholars are also making use of the unplugged approach, as stated in a systematic literature review that studied 125 papers focused on CT [18]. Similar conclusions are reached in a survey on how to teach Computing [30], where 13% of 357 participating in-service teachers affirm that they use unplugged activities in their computer science lessons. Nonetheless, while the effectiveness of computer programming to foster the development of CT skills is being widely investigated [22], this is not the case for the unplugged approach.

Most of experiences using unplugged activities aim to foster learners' interest in computer science. Using questionnaires and interviews, the effect of the CS unplugged activities on middle-school students' views about computer science is examined in [31]. The results show that "although students generally understood what CS is, they perceived the computer as the essence of CS and not primarily as a tool, contrary to the intention of the activities". With similar goals and results, the CS unplugged program was implemented as part of a one-year outreach program for high school students aiming to "excite the next generation of undergraduates about pursuing a degree in computer science" [15]. The findings show that the program had no impact on learners' perceived content understanding nor on their attitudes towards computer science.

Different results are achieved in [20], though, where a group of researchers visit several fourth grade classes aiming to increase interest in computer science making use of CS unplugged activities. The results, based on pre-tests and post-tests, show improved confidence and interest in both computer science and mathematics. Positive results are also found in [14], which summarizes the work performed in 26 different schools for a total of 14,040 hours of classes using unplugged activities. This exploratory study concludes that CT unplugged lessons are a valuable alternative to regular, on-line programming lessons.

The use of the unplugged approach for teacher training has been studied as well. A series of workshops were organized to explore the effectiveness of unplugged methods to introduce educators to computer science topics [12]. The evaluation, based on surveys, "suggests that unplugged activities make for an inspiring and fun session for teachers that they also find useful, interesting and confidence building". In a similar vein, [11] describes how unplugged activities embedded in stories can be used to teach CT ideas. Specifically, the paper presents two examples, "one based on the problem of helping people with locked-in syndrome communicate, the second based around magic tricks". After a 2-hour professional development workshop for teachers, attendants provide positive feedback, 100% of them stating that the workshop had given them useful ideas for the classroom.

Most of the afore reviewed investigations focus on measuring participants' enthusiasm and interest for computing, but there is no assessment on whether participants develop their CT skills with unplugged activities. This is exactly the goal of interventions with middle schools students using an unplugged curriculum [25] [32]. The results support the hypothesis that students do learn CT skills from unplugged activities at least as effective as when following more conventional approaches.

Campos et al. [9] used a CT quiz, which consists of four questions about abstraction, correlation, and codification, to measure students' CT skills before and after the implementation of CT unplugged activities from the CS Unplugged Book. The results, however, were not conclusive.

The review of the literature, hence, highlights that there is a need for more empirical research providing evidence on the usefulness of unplugged activities to develop CT skills, especially when it comes to its use in primary schools. Consequently, in this paper, we try to shed some light on this matter.

3 METHOD

In this section, we describe the sample in our research, and how participants were divided into two different groups-conditions: the experimental group-condition and the control group-condition. Then, we present the instrument used for assessing the CT skills of the participants from both conditions, with a pre-test and a post-test. The pedagogical materials containing the unplugged activities taken by the experimental group along the teaching sessions are then explained. Finally, we report the procedure followed in our quasi-experiment.

3.1 Participants

The valid sample of our quasi-experiment, that is, the set of individuals who were assessed both in the pre-test and post-test time, is composed of 73 students enrolled in 5th and 6th grade (10-12 years old) from two different public primary schools located in Madrid (Spain). The distribution of the sample regarding school, grade, gender, and condition, is presented in Table 1.

[1] CS Unplugged Book: http://csunplugged.org/

Table 1: Distribution of the valid sample (n=73) by grade, age, condition (column Cond), and gender. Possible conditions are: E for 'Experimental' and C for 'Control'.

	Grade	Age	Cond	Gender		Total
				Boys	Girls	
School A	5th	10-11 y.o.	C	10	13	23
			E	10	10	20
School B	6th	11-12 y.o.	C	6	8	14
			E	9	7	16
Total				35	38	73

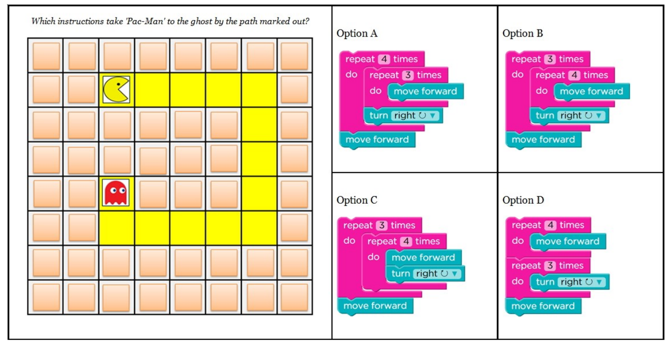


Figure 1: CT Test, item #8 ('maze'): loops 'repeat times' (nested); 'visual blocks'; 'sequencing'.

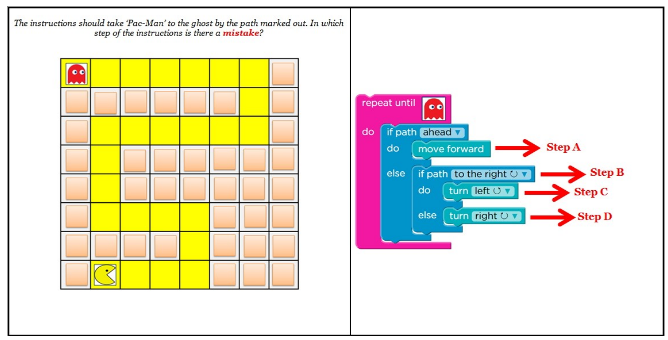


Figure 2: CT Test, item #16 ('maze'): loops 'repeat until' + if/else conditionals (nested); 'visual blocks'; 'debugging'.

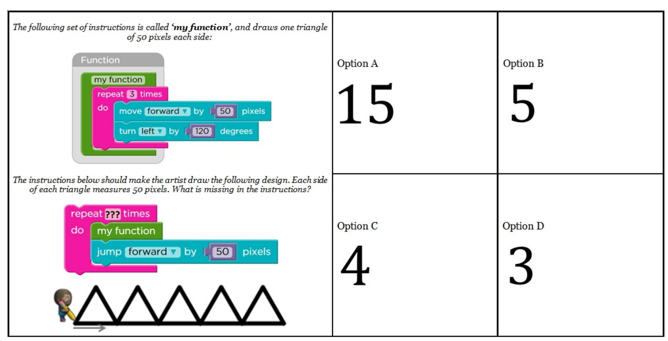


Figure 3: CT Test, item #26 ('canvas'): loops 'repeat times' + simple functions; 'visual blocks'; 'completing'.

3.2 Instrument and Materials

3.2.1 Computational Thinking Test. The Computational Thinking Test (CT Test) [26, 27, 29] was the instrument used to assess the level of CT in the participants in our research. The CT Test measures "the ability to formulate and solve problems by relying on fundamental concepts of computation (i.e., sequences, loops, conditionals, functions, and variables), and using the inherent logic of computer programming". All the items that assemble the test involve, to a greater or lesser extent, the four-pillar cognitive processes of CT: decomposition, pattern recognition, abstraction and algorithmic design. Thus, when a student tries to solve an item (e.g., item #8, see Figure 1), he/she must: break down the steps that the Pac-Man should follow; recognize the visual patterns on the marked path (e.g, in the item #8 there is a repeated pattern that consists of advancing four squares and then turning to the right); abstract the core elements of the problem and ignore the irrelevant details (e.g., such as the colour of the path or the features of the characters); and design an algorithm to solve the problem, which involves some computational concepts (e.g., in item #8, nested loops must be used along the algorithmic design).

The CT Test was selected for our research because of its precise (although necessarily reductionist) operational definition of CT, which may shed some light on the controversy surrounding this often blurry construct [17] [18]. The CT Test was also elected due its quantitative and aptitudinal approach, and because it has already undergone a rigorous validation process, which has stated its content validity [27], criterion validity [29], and convergent validity [26].

Overall, the psychometric studies of the CT Test support that this test is reliable ($\alpha \approx .80$) and valid for assessing the level of CT in students from 10 to 16 years old. The CT Test is composed of a set of 28 multiple choice items with four answer options (only one correct), and it is created and executed on Google Forms technology, being available therefore on virtually any device[2]. Examples of CT Test items are shown in Figure 1, Figure 2 and Figure 3.

3.2.2 Materials for Computational Thinking Unplugged. Most of the pedagogical materials about the unplugged activities taken by the experimental group have been created by the authors, while some were adapted and translated to Spanish from the "Hello Ruby" book [21] and the "Code Master" board game [13]. Some of the activities are presented in Table 2, and most of them are available in the "Computacional" website[3].

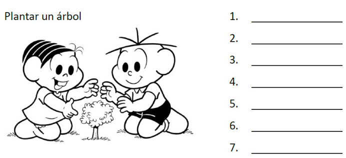
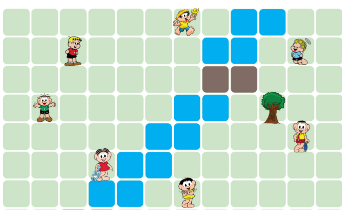
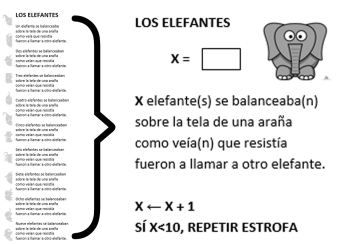
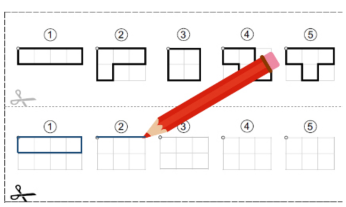
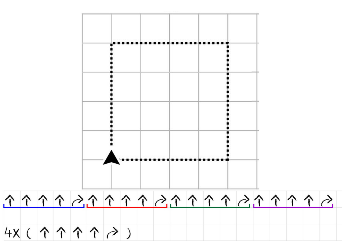
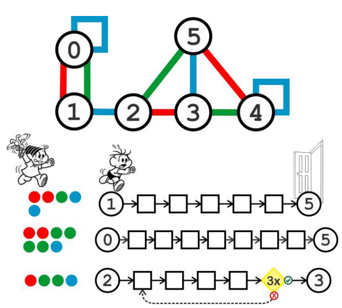
3.3 Procedure

Students in the 5th and 6th grade from two public schools in Madrid (Spain) were invited to participate in the research as part of their regular classes during the second semester of 2016 and the first semester of 2017. We respected the existing grouping of the subjects

[2] A sample copy of the CT Test is available at: https://goo.gl/5O06Oh

[3] http://www.computacional.com.br/atividades/espanhol/

Table 2: Six examples of activities performed by the children

Activity	Explanation	Main Pillars
	**"Decomposition" activity:** Students had to break down many problems (e.g. Plant a tree) identifying all the steps necessary to solve it. Other examples were: Wash Hands, Prepare breakfast, Take an elevator, Tie a shoe, etc.	Decomposition Algorithms
	**"Monica's Map" activity:** A map with many characters is shown to the students and they have to find the shortest route between them using only up, down, left and right arrows (→, ←, ↑, and ↓). On a second moment, they should use multipliers (i.e. → → → → → = 5x→) to write down the solutions.	Pattern Recognition Algorithms
	**"Elephants" activity:** uses a popular students song as exemplification of how a song can turn to an algorithm. In this particular song, the repetition, variables, and conditionals are worked through the increase of the amount of the elephants. Every verse had an increase of the variable until it reached a number equal or bigger than 10.	Abstraction Pattern Recognition Algorithms
	**"Tetris" activity:** some drawings of Tetris pieces are presented to one of the students who gives instructions to its partner. The student who got the upper part of the paper had to hide the images from the partner so it would be possible only to hear the instructions without looking to the answers. The instructions are limited to "start", "up", "down", "left", "right", and "stop". No other words can be used to describe how the figure is drawn.	Pattern Recognition Algorithms
	**"Repetition Drawing" activity:** allows the students to understand the use of repetitions on Tetris-like figures. In this case, the students need to use instructions based on the perspective of the direction of the arrow and try to use the most amount of multipliers in their command. Differently from the "Tetris" activity, the students do it individually and only the use of turn left, turn right and forward are available (↑, ↰, and ↷). The pillars of abstraction, pattern recognition and algorithm are mainly developed.	Decomposition Abstraction Pattern Recognition Algorithms
	**"Monica's Automata":** The last activity is a simpler remake of the Code Master board game developed by the ThinkFun company. In this activity the student is supposed to find a route between two nodes using the allowed colors for each path. All the colors had to be used, leaving no blank spaces. The number located on the left side is the start point and on the right side the finish point.	Decomposition Abstraction Algorithms

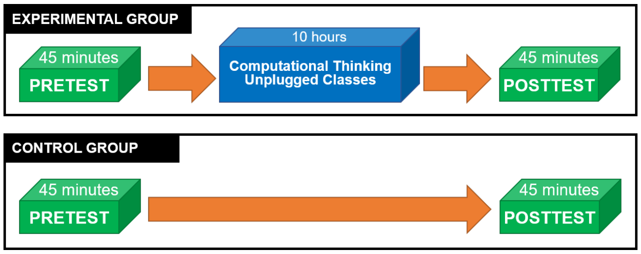


Figure 4: Stages and groups of the project

in their natural classrooms for the assignment of the experimental and control conditions. In other words, the individuals were not randomly assigned to the conditions, so that a quasi-experiment was performed.

For the CT Test collective administration in pre-test time (week #1), none of the students had prior formal programming experience. The test was performed in the school's computer lab. After some students had finished the test, we kept them busy so that they do not distract those students still taking the test.

During the next five weeks, lessons involving CT unplugged activities were administered by the researchers once a week to the experimental group. At first, the schools allowed the researchers to use only one hour per week, but after observing the high motivation of the students and the approval of the teacher, the schools allowed to double the time per week. So, a total of 10 hours of CT unplugged sessions were given. Meanwhile, the control group did not receive any intervention from the researchers.

On average, it was possible to implement two activities per session. On week #7, students from both groups were invited again to take the CT Test in the same way as described before. Therefore, six weeks elapsed between the pre-test and the post-test, which is a sufficient time to avoid the undesirable 'memory-effect' of using an identical set of items at both administrations. A diagram of all the steps of the research is depicted in Figure 4.

All answers by students to the CT Test were stored and available to preview, convert and download on Google Spreadsheets. Answers were then imported and analyzed with the 24^{th} version of IBM SPSS (Statistical Package for the Social Sciences).

4 RESULTS AND DISCUSSION

This section presents and discusses our findings from a double point of view. On the one hand, we report the quantitative results from our quasi-experiment, which intends to answer the following research question: Did the unplugged activities improve the CT skills of the students? On the other hand, we complement the afore mentioned 'hard' results with a qualitative approach, including in the discussion the informal observations of the researchers during the unplugged activities and the CT Test administrations.

4.1 Quantitative Results: Performance in the CT Test

The Table 3 shows the summary quantitative results of our quasi-experiment for the entire valid sample. As it can be seen, the control group had not a statistical significant improvement in the CT Test score between the pre-test and the post-test (t= 1.128; p(t)= .267 > .05); the effect size of the improvement in the control group was d=.17 [24], that can be considered as 'no effect' at all [10]. Conversely, a statistical significant pre-post improvement in the CT Test score is found in the experimental group (t=4.431; p(t)= .000 < .001), which involves a 'large' effect size (d=.80). These results are depicted in Figure 5.

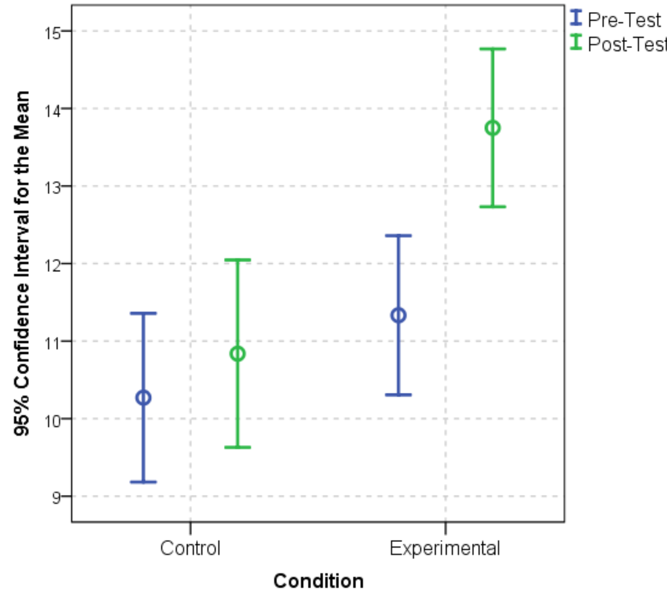


Figure 5: Error bars with the 95% confidence intervals for the means of the CT Test Score for both groups-conditions, and in pre-test and post-test times.

As it can also be seen in Figure 5, there were not statistically significant differences in the CT Test score between the control group and the experimental group at the time of pre-test (t= 1.441; p(t)= .154 > .05). This result indicates that both groups were initially equivalent at the beginning of the quasi-experiment, which is desirable in this type of research design. Conversely, statistically significant differences were found between the control group and the experimental group after our intervention on the latter. (t= 3.730; p(t)= .000 < .001).

In order to test the overall statistical significance of our quasi-experiment, we perform an analysis of covariance (ANCOVA), which checks the differences between control and experimental groups in post-test time taking into account the differences, if any, in pre-test time. The ANCOVA results are statistically significant (F(1,72)=11.690; p(F)=.001 < .01), in favor of the experimental group, with an associated global effect size of our quasi-experiment d=.59 [23], which can be considered in the 'zone of desired effects' to affirm the effectiveness of an educational intervention [10]. Furthermore, this global value is very similar to that found for the CT Test score in a recent and analogous quasi-experiment performed with middle school students who took a 12-weeks Code.org course [28], where a global d=.62 was reported.

Table 3: Summary of quantitative results regarding performance in the CT Test for the entire sample

		Mean	N	SD	Student's t	pre-post d	ANCOVA F	Global d
Control	Pre-test	10.27	37	3.263	1.128	0.17	11.690^{**}	0.59
	Post-test	10.84	37	3.625				
Experimental	Pre-test	11.33	36	3.033	4.431^{***}	0.80		
	Post-test	13.75	36	3.008				

*** p-value < .001; ** p-value < .01; * p-value < .05

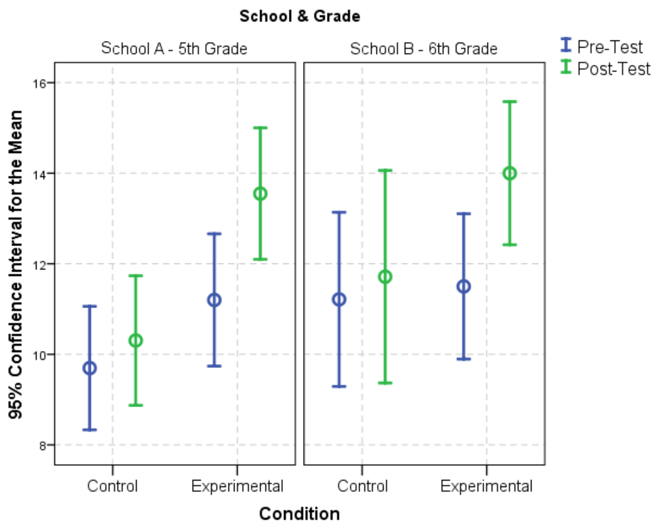


Figure 6: Error bars with the 95% confidence intervals for the means of the CT Test Score, split by school and grade, for both groups-conditions, and in pre-test and post-test times.

We consider that these findings have two more additional implications. Firstly, they support the assertion that the CT Test is valid and sensitive to detect improvements in the CT skills of the students, not only after taking on-line coding courses such as the ones of Code.org [28], but also after receiving CT unplugged activities. Secondly, our results give evidence that the size of the improvement is similar after both types of interventions; this fact might guide future curriculum decisions of teachers and policy makers.

When we split our analysis regarding school and grade (Table 4, Figure 6), we obtain results that globally replicate those found in the entire sample. Furthermore, these segmented results show that the CT Test score seems to increase consistently, not only due to intervention, but also due to age (although this increase regarding age is not statistically significant in our quasi-experiment). Hence, it might be hypothesized that the performance on the CT Test tends to increase as it does the grade. This result would be consistent with the assumption that CT is mainly a problem-solving ability that it should be therefore linked to the cognitive development and maturity of the subjects [1], and it was already found during our large validation study of the CT Test [29]

Overall, our results permit us to answer the research question. It has been demonstrated through a quasi-experimental research design that our set of CT unplugged activities improve the CT skills of the students as measured by the CT Test.

4.2 Qualitative Results: Performance along the Unplugged Activities

As mentioned in subsection 3.3, the schools initially allowed the researchers to use only one hour per week for the unplugged activities; but after watching the motivation of the students, the teachers asked to double the time per week. It was surprising to the researchers because the principals of the schools emphasized at the beginning of the quasi-experiment that it would not be possible. Many notes were taken while the activities were conducted at the schools. Most annotations were related to minor adjustments or corrections of the instructions and small tweaks to better understand the activities. Some of the relevant notes describing qualitative observations of the teaching-learning process are pointed out below. Please see Table 2 as reference.

- The "Monica's Decomposition" activity was the first exercise the groups carried out after the pre-test. The students could not quite understand what they were supposed to do because they were not used to decompose problems. After solving the first two questions as an example, they were able to finish the other ones. When everybody was finished, the researcher read some answers and dramatized the movements to the others students. Many "bugs" were encountered in their algorithm and solved by the students themselves.
- "Monica's Map" activity had an excellent acceptance by the students and it was easy to perform. Some students finished the activity in few minutes, and others took a long time to conclude it. Most students had a hard time finding the path from one point to another in the map and had to fix what they had done before. Many students also did not take the shortest path between two points and a correction was necessary.
- The "Elephants" activity was one of the most cheerful exercises because it involved several choruses and code reading/processing. Since the song was made for small children, the researcher felt that some students from the 6th grade felt uncomfortable with the song. It was the most creative and attractive way found to teach variables to students, and it was possible to achieve the objective.
- During the "Tetris" activity, the students had the opportunity to sit in pairs. Many mistakes happened when the students started the first drawing and errors were getting less often on the following challenges. The instructions

Table 4: Summary of quantitative results regarding performance in the CT Test, split by school and grade

			Mean	N	SD	Student's t	pre-post d	ANCOVA F	Global d
School A (5th Grade)	Control	Pre-test	9.70	23	3.154	-.916	0.19	7.804**	0.55
		Post-test	10.30	23	3.309				
	Experimental	Pre-test	11.20	20	3.122	-3.487**	0.75		
		Post-test	13.55	20	3.103				
School B (6th Grade)	Control	Pre-test	11.21	14	3.332	-.633	0.15	3.497~	0.63
		Post-test	11.71	14	4.065				
	Experimental	Pre-test	11.50	16	3.011	-2.725*	0.83		
		Post-test	14.00	16	2.966				

*** p-value < .001; ** p-value < .01; * p-value < .05; ~ p-value < .10

were not respected many times, and the investigator had to step in.

- During the "Repetition Drawing", many students had difficulties to understand the position and direction in the perspective of the arrow. The exercises had to be explained several times until they understood the difference between this exercise and the "Monica's Map" moving strategy. The best way to make them better understand was standing up and to walk/turn according to the instructions they wrote on the paper.
- "Monica's Automata", which is based on Code Master board game, was the most motivating activity, because it involved several steps (cut, paste and strategize). Since the exercises have more than one correct answer, the students enjoyed very much discussing about the diverse possibilities.

5 LIMITATIONS AND THREATS TO VALIDITY

Some limitations and threats to validity of our research can be pointed out. Firstly, the CT Test has some limitations, since it is heavily focused on computational concepts, only partially covers computational practices, and ignores computational perspectives [8]. Moreover, the CT Test has a (deliberately) reductionist conception of CT, which puts over-emphasis on path-finding algorithms. Secondly, most of the unplugged activities carried out along the research might be considered as excessively and artificially aligned with the items of the CT Test. Therefore, if a different set of unplugged activities had been used, we would probably have obtained different results. Finally, the small size of the sample should be noted (N < 120), in order to consider the limited generalization power of our results.

6 CONCLUSIONS AND FURTHER RESEARCH

This paper presents a quasi-experiment carried out in two primary schools in Spain aiming to develop students' CT skills through a series of unplugged activities. The students were divided into two groups in each of the schools; the experimental groups were the ones who participated in the unplugged class, while the control groups did not take those lessons. The results show that the CT skills of the students in the experimental groups significantly increased after the intervention, while this was not the case for the control groups. Consequently, these findings provide empirical evidence about the effectiveness of the unplugged approach to develop CT skills. They also contribute to reaffirm CT as a cognitive variable, which mainly consists in a problem-solving ability/process whose development can be disconnected from computer programming [36] [37].

It must be taken into account that these results were achieved after just 10 hours of unplugged activities led by a researcher who is not a native Spanish speaker, and that the effect size found is very similar to the one detected in a previous investigation after 12 weeks of programming training in the Code.org platform [28], which highlights the real impact that the unplugged lessons had in the development of CT of participants.

Nevertheless, even if the unplugged activities can be a good resource for introducing students into CT, it is apparent that this approach has limitations and, therefore, further research is necessary to identify the point at which the unplugged approach loses its effectiveness and the use of computing devices is required to keep on developing CT skills. Some investigations are already merging the two approaches and allowing the students to migrate from unplugged to plugged activities [16] [19] [2] in a smoother pace.

Aiming to broaden the sample and replicate the experiment in a different country, at the moment of writing this paper a new research is being carried out in Brazilian schools. The findings of these new interventions will allow us to state stronger conclusions regarding the effectiveness of the unplugged approach as a resource to develop CT skills, as well as to identify potential similarities and differences between countries.

ACKNOWLEDGMENTS

This work was partially supported by the SMART^2 Project and by the Brazilian Ministry of Education (MEC). The work has also been funded in part by the Region of Madrid under project"eMadrid - Investigación y Desarrollo de tecnologías para el e-learning en la Comunidad de Madrid (S2013/ICE-2715)". The authors are very thankful to the teachers and pupils of CEIP República de Ecuador school and CEIP Lope de Vega school (Madrid, Spain). Our gratitude to Yucnary Torres who kindly helped the foreign researcher. We are also very thankful to Estúdios Mauricio de Souza S.A. and ThinkFun Inc. for expressly allowing the use of their creations in the activities.

REFERENCES

[1] Phillip L. Ackerman and Eric L. Rolfhus. 1999. The locus of adult intelligence: Knowledge, abilities, and nonability traits. *Psychology and Aging* 14, 2 (1999), 314–330. https://doi.org/10.1037/0882-7974.14.2.314

[2] Ashish Aggarwal, Christina Gardner-McCune, and David S. Touretzky. 2017. Evaluating the Effect of Using Physical Manipulatives to Foster Computational Thinking in Elementary School. ACM Press, 9–14. https://doi.org/10.1145/3017680.3017791
[3] Association for Computing Machinery. 2003. *A Model Curriculum for K-12 Computer Science: Final Report of the ACM K-12 Task Force Curriculum Committee.* ACM, New York. OCLC: 907036381.
[4] Anja Balanskat and Katja Engelhardt. 2015. Computing our future: Computer programming and coding - Priorities, school curricula and initiatives across Europe. (Oct. 2015). http://www.eun.org/c/document_library/get_file?uuid=3596b121-941c-4296-a760-0f4e4795d6fa&groupId=43887
[5] Tim Bell, Ian H. Witten, and Mike Fellows. 1998. *Computer Science Unplugged... - Off-line activities and games for all ages (draft)* (1 ed.).
[6] Stefania Bocconi, Augusto Chioccariello, Giuliana Dettori, Anusca Ferrari, Katja Engelhardt, Panagiotis Kampylis, and Yves Punie. 2016. *Developing Computational Thinking in Compulsory Education - Implications for policy and practice.* Technical Report. Publications Office of the European Union. https://doi.org/10.2791/792158
[7] Christian Brackmann, Dante Barone, Ana Casali, Rafael Boucinha, and Susana Munoz-Hernandez. 2016. Computational thinking: Panorama of the Americas. IEEE, 1–6. https://doi.org/10.1109/SIIE.2016.7751839
[8] Karen Brennan and Mitchel Resnick. 2012. New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association.* Vancouver, Canada, 1–25. http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf
[9] Gleider Campos, Simone Cavalheiro, Luciana Foss, Ana Pernas, Clause Piana, Marilton Aguiar, André Du Bois, and Renata Reiser. 2014. Organização de Informações via Pensamento Computacional: Relato de Atividade Aplicada no Ensino Fundamental. *20o. Workshop de Informática na Escola (WIE 2014)* (2014).
[10] Jacob Cohen. 1988. Statistical power analysis for the behavioral sciences Lawrence Earlbaum Associates. *Hillsdale, NJ* (1988), 20–26.
[11] Paul Curzon. 2013. cs4fn and computational thinking unplugged. In *Proceedings of the 8th Workshop in Primary and Secondary Computing Education.* ACM, 47–50.
[12] Paul Curzon, Peter W McOwan, Nicola Plant, and Laura R Meagher. 2014. Introducing teachers to computational thinking using unplugged storytelling. In *Proceedings of the 9th Workshop in Primary and Secondary Computing Education.* ACM, 89–92.
[13] Mark Engelberg. 2015. Code Master Programming Logic Game, Thinkfun Inc. (2015).
[14] Hylke Faber, Menno Wierdsma, Richard Doornbos, Jan Salvador van der Ven, and Kevin de Vette. 2017. Teaching Computational Thinking to Primary School Students via Unplugged Programming Lessons. *Journal of the European Teacher Education Network* 12 (2017).
[15] Yvon Feaster, Luke Segars, Sally K Wahba, and Jason O Hallstrom. 2011. Teaching CS unplugged in the high school (with limited success). In *Proceedings of the 16th annual joint conference on Innovation and technology in computer science education.* ACM, 248–252.
[16] Anna Gardeli and Spyros Vosinakis. 2017. Creating the Computer Player: An Engaging and Collaborative Approach to Introduce Computational Thinking by Combining 'Unplugged' Activities with Visual Programming. *Italian Journal of Educational Technology* 1 (July 2017). https://doi.org/10.17471/2499-4324/910
[17] S. Grover and R. Pea. 2013. Computational Thinking in K-12: A Review of the State of the Field. *Educational Researcher* 42, 1 (Jan. 2013), 38–43. https://doi.org/10.3102/0013189X12463051
[18] Filiz Kalelioglu, Yasemin Gülbahar, and Volkan Kukul. 2016. A Framework for Computational Thinking Based on a Systematic Research Review. *Baltic Journal of Modern Computing* 4, 3 (2016), 583.
[19] Lorenz Klopfenstein, Andiy Fedosyeyev, and Alessandro Bogliolo. 2017. BRINGING AN UNPLUGGED CODING CARD GAME TO AUGMENTED REALITY. 9800–9805. https://doi.org/10.21125/inted.2017.2327
[20] Lynn Lambert and Heather Guiffre. 2009. Computer science outreach in an elementary school. *Journal of Computing Sciences in colleges* 24, 3 (2009), 118–124.
[21] Linda Liukas. 2015. *Hello Ruby: adventures in coding.* Feiwel & Friends.
[22] Sze Yee Lye and Joyce Hwee Ling Koh. 2014. Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior* 41 (2014), 51 – 61. https://doi.org/10.1016/j.chb.2014.09.012
[23] Scott B. Morris. 2008. Estimating Effect Sizes From Pretest-Posttest-Control Group Designs. *Organizational Research Methods* 11, 2 (April 2008), 364–386. https://doi.org/10.1177/1094428106291059
[24] Scott B. Morris and Richard P. DeShon. 2002. Combining effect size estimates in meta-analysis with repeated measures and independent-groups designs. *Psychological Methods* 7, 1 (2002), 105–125. https://doi.org/10.1037/1082-989X.7.1.105
[25] Brandon Rodriguez, Kennicutt Stephen, Cyndi Rader, and Tracy Camp. 2017. Assessing Computational Thinking in CS Unplugged Activities. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education.* ACM, Seattle, Washington, USA, 501–506. https://doi.org/10.1145/3017680.3017779
[26] Marcos Román-González, Jesús Moreno-León, and Gregorio Robles. 2017. Complementary Tools for Computational Thinking Assessment. In *Proceedings of International Conference on Computational Thinking Education (CTE 2017)*, S. C Kong, J Sheldon, and K. Y Li (Eds.). The Education University of Hong Kong, 154–159. http://www.eduhk.hk/cte2017/doc/CTE2017Proceedings.pdf
[27] Marcos Román-González. 2015. Computational Thinking Test: Design Guidelines and Content Validation. In *Proceedings of the 7th Annual International Conference on Education and New Learning Technologies (EDULEARN 2015).* IATED, Barcelona, Spain, 2436–2444. https://doi.org/10.13140/RG.2.1.4203.4329
[28] Marcos Román-González. 2016. *Codigoalfabetización y Pensamiento Computacional en Educación Primaria y Secundaria: Validación de un Instrumento y Evaluación de Programas.* Ph.D. Dissertation. Universidad Nacional de Educación a Distancia, Madrid, Spain.
[29] Marcos Román-González, Juan-Carlos Pérez-González, and Carmen Jiménez-Fernández. 2017. Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test. *Computers in Human Behavior* 72 (July 2017), 678–691. https://doi.org/10.1016/j.chb.2016.08.047
[30] Sue Sentance and Andrew Csizmadia. 2015. Teachers' perspectives on successful strategies for teaching Computing in school. *Paper presented at IFIP TCS* 2015 (2015).
[31] Rivka Taub, Mordechai Ben-Ari, and Michal Armoni. 2009. The effect of CS unplugged on middle-school students' views of CS. *ACM SIGCSE Bulletin* 41, 3 (2009), 99–103.
[32] Renate Thies and Jan Vahrenhold. 2013. On plugging "unplugged" into CS classes. ACM Press, 365. https://doi.org/10.1145/2445196.2445303
[33] R. Unnikrishnan, N. Amrita, Alexander Muir, and Bhavani Rao. 2016. Of Elephants and Nested Loops: How to Introduce Computing to Youth in Rural India. ACM Press, 137–146. https://doi.org/10.1145/2930674.2930678
[34] Peter Wallet. 2014. *ICT in Education in Asia: A comparative analysis of ICT integration and e-readiness in schools across Asia.* Technical Report. UNESCO. https://doi.org/10.15220/978-92-9189-148-1-en
[35] Peter Wallet. 2015. *ICT in Education in Sub-Saharan Africa: A comparative analysis of basic e-readiness in schools.* Technical Report. UNESCO. https://doi.org/10.15220/978-92-9189-178-8-en
[36] Jeannette M. Wing. 2006. Computational thinking. *Commun. ACM* 49, 3 (March 2006), 33. https://doi.org/10.1145/1118178.1118215
[37] J. M Wing. 2008. Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 366, 1881 (Oct. 2008), 3717–3725. https://doi.org/10.1098/rsta.2008.0118