DOI: 10.33965/celda2019_201911L011
16th International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2019)

INVESTIGATING K-2 STUDENTS' COMPUTATIONAL THINKING SKILLS DURING A PROBLEM-SOLVING ACTIVITY ABOUT THE WATER CYCLE USING EDUCATIONAL ROBOTICS

Charoula Angeli, Eftychia Xerou and Maria Nicolaou
University of Cyprus, Cyprus

ABSTRACT

The main purpose of the study was to investigate K-2 children's development of computational thinking skills through an activity with Blue-Bot, a floor programmable robot. Twenty children between six and seven years old participated in the study. Children were engaged in problem-solving activities with the Blue-Bot to learn how to program it and about the water cycle. The results of the study reported statistically significant learning gains between the initial and final assessment of children's computational thinking skills as they were measured holistically with a rubric that was developed for the purposes of this study. In addition, the results showed that learning with robotics activities was an effective way to teach children about the water cycle.

KEYWORDS

Computational Thinking, Robotics, Blue-Bot, Science, Young Children

1. INTRODUCTION

Despite the fact that currently there is not one unanimous definition of computational thinking, after a systematic examination of what is currently known in the literature, Grover and Pea (2013) concluded that researchers have come to accept that computational thinking is a thought process that utilizes the elements of abstraction, generalization, decomposition, algorithmic thinking, and debugging (detection and correction of errors). Abstraction is the skill of removing characteristics or attributes from an object or entity in order to reduce it to a set of fundamental characteristics (Wing, 2011). While abstraction reduces complexity by hiding irrelevant detail, generalization reduces complexity by replacing multiple entities which perform similar functions with a single construct (Selby, 2012). For example, programming languages provide generalization through variables and parameterization. Abstraction and generalization are often used together as abstracts are generalized through parameterization to provide greater utility. Decomposition is the skill of breaking complex problems into simpler ones (Wing, 2008; National Research Council, 2010). Algorithmic thinking is a problem-solving skill related to devising a step-by-step solution to a problem and differs from coding (i.e., the technical skills required to be able to write code in a programming language) (Selby, 2012). Additionally, algorithmic notions of sequencing (i.e., planning an algorithm, which involves putting actions in the correct sequence), and algorithmic notions of flow of control (i.e., the order in which individual instructions or steps in an algorithm are evaluated) are also considered important elements of computational thinking (Lu & Fletcher, 2009). Debugging is the skill to recognize when actions do not correspond to instructions, and the skill to fix errors (Bers et al., 2014). For the purposes of the study reported herein, the elements of algorithmic thinking, sequencing, decomposition, and debugging are of particular interest and constitute the main areas of the authors' research investigation.

In regards to teaching computational thinking skills during the last decade, the research community has embraced educational robotics with genuine enthusiasm as an approach for teaching computational thinking to young students (Stoeckelmayr et al., 2011; Bers, 2010; Bers et al., 2014; Kazakoff, Sullivan, & Bers, 2013; Alimisis & Kynigos, 2009; Benitti, 2012; Bredenfeld, Hofmann, & Steinbauer, 2010; Johnson, 2003; Botički, Pivalica, & Seow, 2018). In this study, the authors used the Blue-Bot, shown in Figure 1, to teach children computational thinking skills.

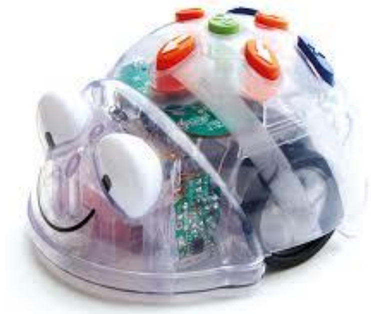

Figure 1. Blue-Bot

Specifically, the study herein sought to investigate the effects of learning with the Blue-Bot on young children's computational thinking skills within the context of two problem-solving scenarios which engaged the children in rich problem-solving activities. The first problem-solving scenario was exploratory in nature in order for the children to learn how to program the Blue-Bot, while the second one was about the water cycle. For this reason, the researchers also examined the effects of learning with the Blue-Bot on children's understanding of the stages of the water cycle. All in all, the researchers sought to answer the following two research questions: (1) Are there any learning gains related to children's understanding of the water cycle? (2) Are there any learning gains related to children's development of computational thinking measured on two different occasions?

The research contributes to the body of knowledge that can be used to inform the teaching of computational thinking skills, and responds directly to calls for more research into how to teach young children computational thinking (Grover & Pea, 2013, 2018; Guzdial, 2008; Lye & Koh, 2014).

2. METHODOLOGY

2.1 Participants

Twenty young children between 6 and 7.5 years old participated in the study. The average age of the participants was 6.68 years (*SD* = .47). Of the 20 participants, 13 were girls and seven were boys. Students were randomly selected from different private and public schools located in urban and rural areas in a southern Mediterranean country. The majority of the students (13) were bilingual and a small number of students (4) were trilingual. Prior to participating in the study, the researchers obtained written consent from the children's parents or guardians. All children had no previous experience with either computers or robotic devices.

2.2 Blue-Bot

The Blue-Bot is a robotic programmable floor device in the shape of a bee. It is transparent and when is working a blue light turns on. Students can program the Blue-Bot by using the directional buttons located on its back or through Bluetooth from a tablet or a mobile phone. For the purposes of the current study students programmed the robotic device by using the directional buttons. The directional buttons can move the Blue-Bot in four different directions, namely, backward (15 cm), forward (15 cm), left (90 degrees), and right (90 degrees). The execution of a command or a sequence of commands starts with the push of the GO button.

The CLEAR button is used to clear the Blue-Bot's memory, which can hold up to 40 commands at a time. It has also a PAUSE button which can pause the function of the robotic device for one second. Blue-Bot can be charged through a USB cable. The Blue-Bot blinks and beeps at the conclusion of each sequence of commands.

2.3 Knowledge Test about the Water Cycle

A knowledge test was developed by the researchers to assess students' conceptual knowledge about the water cycle. The test was used both as a pretest and a posttest. The test consisted of two questions. The first question was a matching question asking the children to match the six words on the first column with the six pictures on the second column. The second question instructed the students to number from one to five the pictures provided so that they represented in the correct order the five stages of the water cycle. Each correct answer was given a score of 1 point, thus the maximum possible score on the first exercise was 6 points. The second question instructed the students to number from one to five the pictures provided so that they represented in the correct order the five stages of the water cycle. The correct placing of all stages was given 1 point to the students. The maximum possible score on the test was 7 points for both questions. The test was administered individually in 5 minutes.

2.4 Problem-Solving Scenarios and Mats

Research participants were engaged in problem-solving tasks using two scenarios and two mats, one mat for each scenario. The mats are shown in Figures 2 and 3. Both scenarios requested children to program the Blue-Bot to perform a certain route on a mat. Sometimes the route involved a number of obstacles that the children needed to find a way to bypass by programming the Blue-Bot to take an alternate route. The first scenario aimed at familiarizing the children with the Blue-Bot by asking them to program it in order to follow certain paths on the first mat to collect different items suitable for a journey, such as, a bag-pack, a bottle of water, a boat, sunglasses and several others in order to prepare the Blue-Bot for his journey on the second mat.

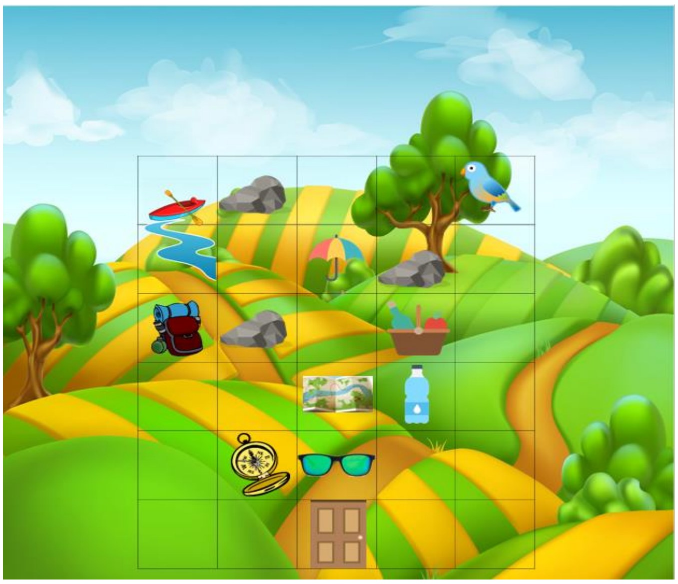

Figure 2. First mat

The second scenario made use of the second mat, shown in Figure 2, and aimed at teaching children about the water cycle by completing a journey through the sun, the clouds etc. This was done through programming the Blue-Bot to undertake a journey consisting of several paths in order to learn about the water cycle.

ISBN: 978-989-8533-93-7

2.5 The Assessment of Computational Thinking

There is currently a dearth of research instruments in the literature for assessing young children's computational thinking. In this study, the researchers developed inductively a rubric for assessing children's computational thinking in a holistic way. The rubric is presented in the results section.

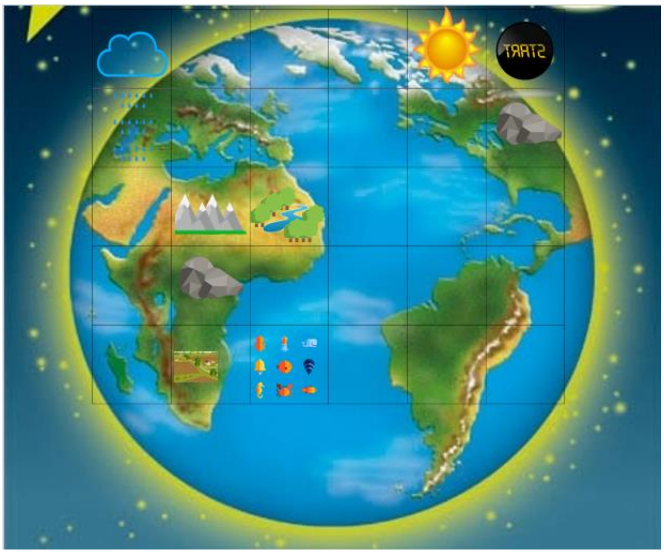

Figure 3. Second mat

2.6 Research Procedure

The research data for this study were collected in two 45-min sessions. Students worked individually in both sessions. There was a two-day elapsed time between the sessions. During the first five minutes of the first session, participants answered the pretest knowledge test about the water cycle. For the remaining 40 minutes, they followed the instructions on the first problem-solving scenario in order to learn how to program the Blue-Bot. Two days later, the children were engaged in the second problem-solving task. During the first 40 min of the second session, they programmed the Blue-Bot to execute a number of paths in order to learn about the water cycle. At the last five minutes of the session, students answered the posttest knowledge test about the water cycle.

3. RESULTS

3.1 Rubric for Assessing Children's Computational Thinking

Children's computational thinking was assessed based on the correctness of their sequences of instructions expressed in Blue-Bot's command language, such as, MOVE FORWARD, TURN LEFT, MOVE BACKWARD, and TURN RIGHT for each problem-solving subtask. In effect, the children were assessed on their sequencing skills taking into consideration previous attempts before succeeding. The rubric also provided a way to evaluate how many failed attempts the children made before succeeding, and whether they used decomposition as a computational thinking strategy. A complete example follows:
X's first attempt (unsuccessful): MOVE BACKWARD-TURN RIGHT-GO
X's second attempt (unsuccessful): MOVE BACKWARD-TURN RIGHT-MOVE FORWARD-GO
X's third attempt (successful): MOVE BACKWARD-TURN RIGHT-TURN RIGHT-MOVE FORWARD-GO

The researchers collected all possible answers for all subtasks over the two problem-solving scenarios and developed a rubric as shown in Table 1. Table 1 shows a holistic assessment of computational thinking taking into consideration the number of attempts students made, and, if decomposition was used as a problem-solving strategy. An interrater reliability for the rubric was calculated between two independent raters and a 100% of agreement was established.

Table 1. Computational Thinking Scoring Rubric

Code	Description	Score
1	Success on 1st attempt without decomposition	22
2	Decomposition in two parts - Success on 1st attempt	21
3	Decomposition in three parts - Success on 1st attempt	20
4	Decomposition in four parts - Success on 1st attempt	19
5	Decomposition in five parts – Success on 1st attempt	18
6	Decomposition in six parts – Success on 1st attempt	17
7	Decomposition in seven parts – Success on 1st attempt	16
8	Decomposition in eight parts – Success on 1st attempt	15
9	Decomposition in nine parts – Success on 1st attempt	14
10	Decomposition in ten parts - Success on 1st attempt	13
11	Success on 2nd attempt without decomposition	12
12	Decomposition in two parts – Success on 2nd attempt	11
13	Decomposition in three parts - Success on 2nd attempt	10
14	Decomposition in four parts - Success on 2nd attempt	9
15	Decomposition in five parts - Success on 2nd attempt	8
16	Decomposition in six parts - Success on 2nd attempt	7
17	Decomposition in eleven parts - Success on 2nd attempt	6
18	Decomposition in four parts – Success on 3rd attempt	5
19	Decomposition in five parts - Success on 3rd attempt	4
20	Decomposition in six parts - Success on 3rd attempt	3
21	Decomposition in 3 parts – Success on 4th attempt	2
22	Decomposition in 2 parts – Success on 5th attempt	1

3.2 Performance on the Knowledge Test

Students' descriptive statistics in regards to their performance on the pretest and posttest knowledge tests about the water cycle were computed. The descriptive statistics showed an average performance on the pretest knowledge test of 5.7 (SD = 1.03), and an average performance on the posttest knowledge test of 6.75 (SD = .72). A paired samples t-test was subsequently performed showing statistically significant differences between pre and posttest knowledge scores, $t\,(19) = 3.68$, $p < 0.01$.

3.3 Holistic Computational Thinking Performance

Descriptive statistics in regards to students' performance on the computational thinking tasks were computed. Students' mean computational thinking performance on the first problem-solving task was found to be 132.23 (SD = 17.85) while students' mean computational thinking performance on the second problem-solving task was found to be 144.90 (SD = 10.41). A paired samples t-test was then computed showing statistically significant learning gains between the first and second assessment of students' computational thinking skills, $t(19) = 3.57$, $p < 0.01$.

4. DISCUSSION AND IMPLICATIONS

In this study, it was hypothesized that the use of robotics activities with the use of a small programmable floor robot named Blue-Bot would be an effective way for developing young children's computational thinking. The hypothesis was confirmed based on the results of the study reporting statistically significant learning gains between the initial and final assessment of children's computational thinking skills. In addition, the authors hypothesized that learning with robotics activities would be an effective way to teach children about the water cycle. This hypothesis was also confirmed as the findings showed statistically significant within-subjects learning gains between the initial and final assessment of children's understanding of the water cycle. The implications of these findings are important, as they provide a robust support for the integration of robotic devices in the curriculum of young children's education. It represents a great initial tool in the hands of the teachers for the use of the specific robotic devices in class. Furthermore, the robotic activities mentioned above provides a strong foundation for additional efforts for the creation of a complete curriculum of teaching computational thinking through robotics and programming for K12. In essence, robotics can be used as an educational technology to reform and enhance the traditional school curriculum so that children can be taught about computational thinking skills that are so much needed for surviving in the 21st century. In addition, these findings show that robotics can be used to teach young students about the water cycle, which constitutes an important component of the science curriculum in the pre-primary and primary education in an alternative way of teaching, through technology. The use of the specific floor robotic device in this lesson transformed completely a traditional method of teaching a specific subject through a technology enhanced lesson that inspired the students to discover and evolve their computational thinking skills. The results of this study can serve as baseline data for future research studies with larger sample sizes. Moreover, the authors explain that due to the fact that this was a one-sample only study no comparisons with other methods can be made at this point. Thus, it is highly recommended that researchers undertake future studies with more than one group of participants so that comparisons with a control group and other approaches can be made.

REFERENCES

Alimisis, D., & Kynigos, C. (2009). Constructionism and robotics in education. *Teacher Education on Robotic-Enhanced Constructivist Pedagogical Methods*, 11-26.

Benitti, F. B. V. (2012). Exploring the educational potential of robotics in schools: A systematic review. *Computers & Education*, *58*(3), 978-988.

Bers, M. U. (2010). The TangibleK robotics program: Applied computational thinking for young children. *Early Childhood Research & Practice*, *12*(2), 1-20.

Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, *72*, 145-157.

Botički, I., Pivalica, D., & Seow, P. (2018). The use of computational thinking concepts in early primary school. In *Proceedings of the International Conference of Computational Thinking Education 2018 (CTE2018).*

Bredenfeld, A., Hofmann, A., & Steinbauer, G. (2010). Robotics in education initiatives in Europe-status, shortcomings and open questions. In *Workshop Proceedings of International Conference on Simulation, Modeling and Programming for Autonomous Robots* (pp. 568-574). Darmstadt: Germany.

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

Grover, S., & Pea, R. (2018). Computational thinking: A competency whose time has come. *Computer Science Education: Perspectives on Teaching and Learning in School*, 19.

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

Johnson, J. (2003). Children, robotics, and education. *Artificial Life and Robotics*, *7*(1-2), 16-21.

Kazakoff, E. R., Sullivan, A., & Bers, M. U. (2013). The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood. *Early Childhood Education Journal*, *41*(4), 245-255.

Lu, J. J., & Fletcher, G. H. (2009). Thinking about computational thinking. In *ACM SIGCSE Bulletin, 41*(1), 260-264. Chattanooga, TN: ACM. Retrieved 24 June, 2015 from URL http://portal.acm.org/citation.cfm?id=1508959&dl=ACM&coll=portal

Lye, S. Y., & Koh, J. H. L. (2014). Review on teaching and learning of computational thinking through programming: What is next for K-12? *Computers in Human Behavior*, *41*, 51-61.

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

Selby, C. C. (2012). Promoting computational thinking with programming. In *Proceedings of the 7th Workshop in Primary and Secondary Computing Education* (pp. 74-77). New York: ACM.

Stoeckelmayr, K., Tesar, M., & Hofmann, A. (2011). Pre-primary children programming robots: A first attempt. In *Proceedings of 2nd International Conference on Robotics in Education* (pp. 185-192). Vienna, Austria: INNOC - Austrian Society for Innovative Computer Sciences.

Wing, J. M. (2008). Computational thinking and thinking about computing. Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences, 366(1881), 3717-3725.

Wing, J. (2011). Research notebook: Computational thinking - What and why? Retrieved 3 March, 2015 from URL http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why