Developing Mathematical Thinking with Scratch

An Experiment with 6th Grade Students

Luis Alberto Calao[1], J. Moreno-León[2(✉)],
Heidy Ester Correa[1], and Gregorio Robles[3]

[1] Institución Educativa Candelaria Hacienda de Lorica,
Santa Cruz de Lorica, Colombia
inge484@yahoo.es
[2] Programamos.es, Sevilla, Spain
jesus.moreno@programamos.es
[3] Universidad Rey Juan Carlos, Madrid, Spain
grex@gsyc.urjc.es

Abstract. One of the latest trends in the educational landscape is the introduction of computer programming in the K-12 classroom to develop computational thinking in students. As computational thinking is not a skill exclusively related to computer science, it is assumed – but not yet scientifically proven – that the problem solving process may be generalized and transferred to a wide variety of problems. This paper presents a research designed to test whether the use of coding in Maths classes could have a positive impact on learning outcomes of students in their mathematical skills. Therefore, the questions we want to investigate in this paper are if the use of programming in Maths classes improves (a) modeling process and reality phenomena, (b) reasoning, (c) problem formulation and problem solving, and (d) comparison and execution of procedures and algorithms. We have therefore designed a quantitative, quasi-experimental experiment with 42 participating 6th grade (11 and 12 years old) students. Results show that there is a statistically significant increase in the understanding of mathematical processes in the experimental group, which received training in Scratch.

Keywords: Computational thinking · Maths · Learning · Coding · Scratch

1 Introduction

One of the latest trends in the educational landscape is the introduction of computer programming in the K-12 classroom to develop computational thinking (CT) in students, a skill defined by Jeannette Wing as one that "involves solving problems, designing systems, and understanding human behaviour, by drawing on the concepts fundamental to computer science" [22]. Although CT is not a skill exclusively related to computer science [4,20], research shows that programming is a very good mechanism for the development of this competence [12].

© Springer International Publishing Switzerland 2015
G. Conole et al. (Eds.): EC-TEL 2015, LNCS 9307, pp. 17–27, 2015.
DOI: 10.1007/978-3-319-24258-3_2

Thus, governments around the world are making computer programming part of their national curriculum. For instance, nine countries in Europe have already included coding into their schools: Bulgaria, Cyprus, Denmark, Estonia, Greece, Ireland, Poland, Portugal and the UK (England) [19].

One of the pillars of CT is, according to the operational definition developed by the Computer Science Teachers Association and the International Society for Technology in Education [3], "generalizing and transferring this problem solving process to a wide variety of problems". Hence, the general research question of this paper is how far CT affects other school subjects. In particular, this study was designed to test whether the use of coding in math classes could have a positive impact on learning outcomes of students in relation to their mathematical skills. Therefore, the specific questions we want to give an answer to in this research paper are following:

1. *Modeling*: Does the use of programming in math classes improve the modeling of processes and reality phenomena?
2. *Reasoning*: Does the use of programming in math classes improve reasoning?
3. *Problem solving*: Does the use of programming in math classes improve problem formulation and problem solving?
4. *Exercising*: Does the use of programming in math classes improve the comparison and execution of procedures and algorithms?

The structure of this paper is as follows: A brief overview of the mathematical skills of Colombian students is presented, and some research studies that have investigated relationships between coding with Scratch, the programming learning environment used in this experiment, in schools and learning are outlined in Sect. 2. Then, in Sect. 3 our work methodology is briefly described. Section 4 presents the results we have obtained from applying our methodology on a small group of Colombian students in the mathematics class. Finally, Sect. 5 contains the conclusions of our research, and some ideas and suggestions for future work are discussed.

2 Background

2.1 Colombian Students and Maths

The results in international tests show that there is still much room for improvement regarding the mathematical skills of Colombian students. In PISA 2012 [13] (Programme for International Student Assessment), which assessed the competencies of 15 year old students in reading, mathematics and science (with a focus on mathematics) in 65 countries and economies, Colombian students scored 376 points in mathematics on average, compared to an average of 494 points in OECD (Organisation for Economic Co-operation and Development) countries. Therefore, Colombia ranked 61st in mathematical competencies.

According to OECD report *Does math make you anxious?* [14], mathematics can "provoke worry, stress and even feelings of powerlessness in some students, and this anxiety towards mathematics is shown to be strongly related to mathematics performance." In order to measure that anxiety, PISA 2012 included questions regarding how students feel when they anticipate having to perform mathematical tasks, when they anticipate their performance in mathematics class, and while they are attempting to solve mathematics problems. The responses revealed that countries where students report higher levels of anxiety were also those where students perform poorer in mathematics. Thus, Colombia was one of the countries with higher levels of anxiety towards maths by students [13].

2.2 Code to Learn with Scratch

The educational use of programming is not new. Back in the 1960s Seymour Papert developed the `Logo` programming language aiming to allow children to use computers to create games, composing music or painting recursive drawings [15]. However, after a few years of success, programming disappeared from the K12 educational landscape because of the problems that students and teachers faced trying to learn the language syntax, among other reasons [16].

Nevertheless, in the last years new visual programming languages, such as Alice, Kodu and especially Scratch [18], have reawakened the interest of the educational community in coding, not as an end in itself, but as a tool to develop other skills and to improve learning outcomes and motivation in students, as Mitchel Resnick, creator of Scratch, argues in *Learn to code, code to learn* [17].

Regarding the usefulness of programming with Scratch as a tool to improve student learning, there is research literature that has a very promising outlook, as coding has been successfully utilized in subjects like mathematics, science, arts, writing or English as a second language, among others. Focusing on mathematics, Lewis and Shah [11] detect correlation between programming quizzes and math tests grades, Ke [8] explains that students showed significantly more positive attitude towards this discipline after the study, while Zavala, Gallardo and García-Ruiz [23] observe improvements in the identification and comparison of numbers, although no gain in relation to the spatial location was detected. However, most of the studies reviewed did not follow basic recommendations to develop research in education [2], and therefore, there is a need to carry out empirical studies using control groups and providing quantitative data to prove the potential of computer programming with Scratch as an educational tool to improve academic outcomes.

Furthermore, there are research studies that analyze the development of problem solving skills while learning to program with Scratch. Most of the articles reviewed confirm that students developed their problem solving skills after the investigation [1,5,9,10,21], while no significant differences were detected in one of the studies [6].

3 Methodology

3.1 Design

This research is a quantitative, quasi-experimental study, which includes a pre-post test design with both experimental and control groups. In this type of research, according to Hernandez, Fernandez and Baptista [7], an independent variable is deliberately manipulated in order to observe its effect and relation to one or more dependent variables through measurements of the subjects before and after treatment application.

3.2 Population

The population that was part of this experiment is composed 42 students of 6th grade of the Candelaria Hacienda school located in the municipality of Lorica, Department of Córdoba of Colombia. This institution is an official primary and secondary school. The sample is intentional and not probabilistic, and it is comprised of 24 students from 6th-1 group, taken as experimental group, and 18 students from 6th-2 group, as control group.

3.3 Data Collection

As an instrument for collecting data, a rubric was elaborated to evaluate students performance and skills in the mathematical processes involved in this investigation: modeling, formulation and problem solving, reasoning and exercising. This rubric was prepared taking into account the conceptual approaches set by the Ministry of National Education of Colombia in relation to the guidelines and basic competence standards in mathematics.

The rubric consists of a set of criteria that evaluates the development of the student in each of the four skills studied in this research. Depending on this criteria, the students gets an evaluation that may be *excellent*, *good*, *satisfactory* and *deficient*.

For instance, students with an excellent performance level in each skill should be able to:

- *Modeling*: The student properly solves all the problems related to the modeling process, in which the detection of variables and relationships among them that establish a mathematical model is required, as well as detecting patterns that are repeated in daily, scientific and mathematical situations, and reconstruct them mentally.
- *Reasoning*: The student properly uses the reasoning process to resolve all situation problems it faces, sensing regularities and relationships, making predictions and guesses, or justifying arguments and reasoning.
- *Problem solving*: The student easily solves problems in situations which require deployment strategies to interpret the statements given, to find results and to verify these results.

– *Exercising*: The student runs easily algorithmic procedures, realizing the concepts on which they rest and recognizing when you can apply a given technical or mathematical operation.

For the application of the rubric two standard test questionnaires were elaborated, one for the pre-test and one for the post-test, which were simultaneously applied to both the control group and the experimental group. Each questionnaire consisted of 16 questions equally distributed among the four skills to evaluate. A copy, both in Spanish and in English, of the rubric and the two questionnaires can be obtained from the replication package of this paper[1].

3.4 Areas of Intervention

The initial stage of the experimental intervention, the experimental group begins with several informal activities on sequences of processes, and then an introduction to the concepts of algorithm and programming is given. In a second stage, students begin to learn to use the Scratch graphical programming environment, conducting educational activities. These activities are initially aimed at basic interactions with the program and then focused on the use of animated dialogues.

In a third stage the use of loops, conditionals and variables is trained. Finally, in a fourth stage, the students test their creativity by programming their own games and simulations making use of images, sounds and movements.

The intervention on the experimental group took place along 3 months. Meanwhile the control group continued the classes using the same kind of methodology and activities they had been using up until that moment.

3.5 Data Analysis

For the data collection, as mentioned earlier, both a pre-test, developed before the intervention, and a post-test, performed after the intervention, were applied to both the control and experimental groups. In each of these tests, a rubric prepared from the conceptual definition of each of the mathematical processes that students are expected to develop in sixth grade of primary education in Colombia are used.

In addition of the overall average of the processes evaluated, the results of the tests offer information regarding the four mathematical processes assessed in this research. A t-test analysis for independent samples is performed comparing the results of the means obtained by the experimental group and the control group in the pre-test and post-test. Similarly, a t-test for related samples is applied to compare the control and experimental groups separately.

[1] http://gsyc.urjc.es/~grex/repro/2015-ectel.

4 Findings

Figure 1 shows the mean and standard deviation of the tests by the control and experimental groups before the intervention. As shown in the figure, the results of the pre-tests are very similar in both groups, which is congruent with the fact that the groups are formed by students of similar characteristics.

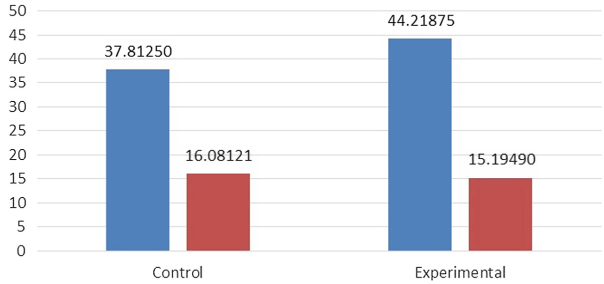

Fig. 1. Pre-test. Mean and deviation for control (left) and experimental (right) groups.

Figure 2 shows the mean and standard deviation of the tests by the control and experimental groups after the intervention. As shown in the figures, there is a remarkable difference in the results of the post-tests, as the mean of the results of the experimental group is 41.56 points above the one of the control group.

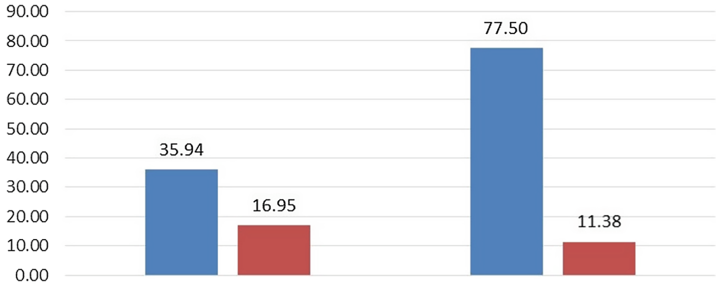

Fig. 2. Post-test. Mean and deviation for control (left) and experimental (right) groups.

Figure 3 offers a comparison of the results obtained for each of the four skills under study for the control and experimental groups.

If the results of each mathematical process before the intervention are compared, we can see that both the mean and the standard deviation are uniform, presenting small differences in the modeling, reasoning and exercising, and only showing a difference of more than 10 points in terms of problem solving where the experimental group shows a significant better performance than the control group.

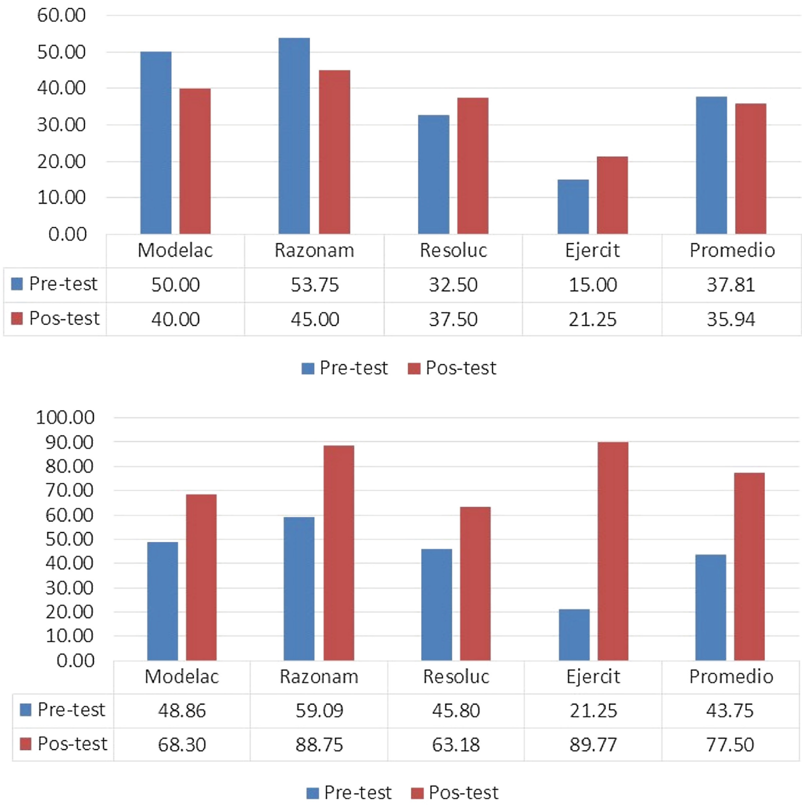

Fig. 3. Comparison. Means obtained for the control (upper) and experimental (lower) groups by mathematical process. *Modelac* stands for *modeling*, *Razonam* for *reasoning*, *Resoluc* for *problem solving*, *Ejercit* for *exercising* and *Promedio* for *average*.

If the results of each mathematical process after the intervention are compared, we can see that there is a significant difference in all the processes in favour to the experimental group. Especially noteworthy are the results of the *exercising* process, with a difference of over 68 points.

While no significant differences are observed in the results of the control group, a uniform gain is shown in the results of the experimental group, where the grades are significantly better in the post-test for all processes, and there is an average increase more than 33.5 points.

It is remarkable that the control group has obtained a lower mean value in the post-test than in the pre-test. While the difference is not significant, it however means that students have not progressed in these skills in the three months of the study. As this students have been attending their regular math classes as usual, this result may hint that (at least some parts of) the math curriculum in schools does not help in developing those skills.

The questions that have been included in the pre and post-test questionnaires are closely connected to the use of mathematics in real-life scenarios, and can be considered as applying mathematics to usual situations. From our results, it seems that math learning at schools is more focused on the internals of mathematics, rather than on acquiring skills to use math-based knowledge. For this type of skill development, our experiment has shown that the use of programming is of great value with a significant increase of the results obtained by learners.

If we look more in detail at the four skills under study, we see that modeling and reasoning are those skills that are more developed with the *traditional* way of learning math. However with *problem solving* and especially with *exercising* there is much room for improvement. From our experiment we have observed that the *exercising* skill in students is particularly increased with the introduction of programming. The mean value for this skill for the experimental group is even the highest among all skills. This finding shows that introducing programming offers students more insight in the comparison and execution of algorithms and procedures than in *traditional* classes, which sounds meaningful as the nature of programming is very related to those activities. In the case of *problem solving*, although the gain in the experimental group is significantly higher then the one for the control group, our results show that the growth is moderate. Further research should be devoted to find out how to further improve this skill, and if programming can be helpful in achieving this goal.

A t-test was performed to measure the average level of mathematical processes in the control and experimental groups in order to assess whether the results are similar or if they have significant differences. The null hypothesis states that there are no significant differences in the sample means.

First, data in Table 1 show that there are no significant differences in the results means of the control group and the experimental group in the pre-test, as the p-value (0.195) is greater than 0.05. However, there are statistically sig-

Table 1. T-test of independent samples in the pre-test and post-test

		Levene test	T-test for equal mean						
		F	Sig	t	Sig. (bilat)	Diff. means	Typ. error	95 % conf. interval	
								Inferior	Superior
Pre-test	Equal var. assumed	0.052	0.821	-1.319	0.195	-6.406	4.857	-16.223	3.410
	Equal var. not as.			-1.308	0.199	-6.406	4.898	-16.343	3.531
Post-test	Equal var. assumed	3.155	.084	-9.246	>0.001	-41.563	4.495	-50.663	-32.462
	Equal var. not as			-8.893	>0.001	-41.563	4.674	-51.125	-32.000

Table 2. Test of related samples. Control and experimental groups.

	Related differences					t	gl	Sig (bilateral)
	Mean	SD	Typical error	95 % confidence interval				
				Inferior	Superior			
Control group	1.875	11.250	2.652	-3.719	7.469	0.707	17	0.489
Exp. group	-33.750	14.315	3.0519	-40.097	-27.403	-11.05	21	>0.001

nificant differences in the results of both groups in the post-test, as the p-value (>0.001) is less than 0.05.

Second, we have applied the t-test to the related samples to study the differences between the pre-test and post-test. Table 2 shows that no statistically significant changes are observed in the control group between the pre-test and post-test , as the p-value (0.489) is grater than 0.05. However in the experimental group there were significant differences between the pre-test post-test, as the p-value (>0.001) is less than 0.05.

5 Conclusions and Further Research

The goal of this research was to analyze the effect of the development of computational thinking through the use of the Scratch visual programming environment in the development of mathematical skills in sixth graders of elementary education, for which a comparison was made between two groups of similar characteristics of the same grade, designating one as control group and the other as experimental group. The latter was the one who received intervention, which consisted in Scratch programming training for three months. Statistical tests were applied to both control and experimental groups before and after the intervention.

The results show that there is a statistically significant gain in the understanding of mathematical knowledge in the experimental group, which received training in Scratch. This therefore leads to the conclusion that the development of computational thinking using the Scratch visual programming environment allows students of primary education to improve their performance in terms of mathematical processes of modeling, reasoning, problem solving and exercising, while, in parallel it also facilitates the generation of a motivating learning environments.

Among the studied skills, we have found that the exercising skill (i.e., the comparison and execution of procedures and algorithms) is the one that is less developed in *traditional* math classes, but is the one that is especially strengthened by programming. Our findings show that modeling and reasoning skills benefit as well significantly from programming. Finally, the problem solving skill also increases, but is the one that has more room for improvement. Future research should focus on how programming may enhance the problem solving skill; some possible solutions could be to address this skill by designing specific programming tasks or by introducing specific methodologies to learn programming.

As a future line of research, we note that during the intervention process we observed that some learners faced some issues in reading and writing when working with lively dialogues. However, students were highly motivated to improve their skills and overcome the difficulties in order to create good Scratch projects. This suggests a possible investigation into how the use of the Scratch visual programming environment can have a positive impact on the development of reading and writing skills.

At this moment the authors are performing a study with more than 40 teachers from three different countries, Spain, Argentina and Ecuador, involving more than 500 students. In this investigation educators from different grades are using computer programming with Scratch to teach several kind of subjects, such as mathematics, literature, English as a second language or social studies, among others. Thus, the results of the research may allow us to measure to what extent the development of computational thinking can have a beneficial impact on the academic outcomes of learners in different disciplines.

Acknowledgments. The work of Jesús Moreno-León and Gregorio Robles has been funded in part under project "eMadrid - Investigación y Desarrollo de tecnologías para el e-learning en la Comunidad de Madrid" (S2013/ICE-2715) funded by the Region of Madrid. The work of Gregorio Robles has been funded in part by the Spanish Government under project SobreSale (TIN2011- 28110).

References

1. Brown, Q., Mongan, W., Kusic, D., Garbarine, E., Fromm, E., Fontecchio, A.: Computer aided instruction as a vehicle for problem solving: Scratch programming environment in the middle years classroom (2013). Accessed on September 22
2. Cohen, L., Manion, L., Morrison, K.: Research Methods in Education. Routledge, 2 ParkSquare, MiltonPark, Abingdon, Oxon OX14 4RN (2007)
3. CSTA, ISTE: Computational thinking, teachers resources. Technical report, CSTA and ISTE (2011). http://www.csta.acm.org/Curriculum/sub/CurrFiles/472.11CTTeacherResources_2ed-SP-vF.pdf
4. Denning, P.J.: The profession of it beyond computational thinking. Commun. ACM **52**(6), 28–30 (2009)
5. Giordano, D., Maiorana, F.: Use of cutting edge educational tools for an initial programming course. In: Global Engineering Education Conference (EDUCON), 2014 IEEE. pp. 556–563. IEEE (2014)
6. Gülbahar, Y., Kalelioğlu, F.: The effects of teaching programming via Scratch on problem solving skills: a discussion from learners perspective. Inform. Educ. Int. J. **13**(1), 33–50 (2014)
7. Hernández Sampieri, R., Fernández Collado, C., Baptista Lucio, P.: Metodología de la Investigación, 5th edn. McGraw-Hill, Mexico (2010)

8. Ke, F.: An implementation of design-based learning through creating educational computer games: a case study on mathematics learning during design and computing. Comput. & Educ. **73**, 26–39 (2014)
9. Lai, A.F., Yang, S.M.: The learning effect of visualized programming learning on 6th graders' problem solving and logical reasoning abilities. In: 2011 International Conference on Electrical and Control Engineering (ICECE), pp. 6940–6944. IEEE (2011)
10. Lai, C.S., Lai, M.H.: Using computer programming to enhance science learning for 5th graders in Taipei. In: 2012 International Symposium on Computer, Consumer and Control (IS3C), pp. 146–148. IEEE (2012)
11. Lewis, C.M., Shah, N.: Building upon and enriching grade four mathematics standards with programming curriculum. In: Proceedings of the 43rd ACM Technical Symposium on Computer Science Education. pp. 57–62. ACM (2012)
12. 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. **41**, 51–61 (2014)
13. OECD: Pisa 2012 results: What students know and can do student performance in mathematics, reading and science (volume i, revised edition, February 2014). Technical report, OECD (2014). http://www.content/book/9789264208780-en
14. OECD: Does math make you anxious? Technical report, OECD (2015). http://www.content/workingpaper/5js6b2579tnx-en
15. Papert, S., Solomon, C.: Twenty things to do with a computer. In: Soloway, E., Spohrer, J.C. (eds.) Studying the Novice Programmer. Lawrence Erlbaum Associates Inc., Hillsdale (1971)
16. Resnick, M.: Point of view: reviving papert's dream. Educ. Technol. **52**(4), 42 (2012)
17. Resnick, M.: Learn to code, code to learn. How programming prepares kids for more than math. EdSurge 8 (2013)
18. Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., et al.: Scratch: programming for all. Commun. ACM **52**(11), 60–67 (2009)
19. Schoolnet, E.: Computing our future. computer programming and coding priorities, school curricula and initiatives across europe. Technical report European Schoolnet (2014). http://www.eun.org/publications/detail?publicationID=481
20. Settle, A., Perkovic, L.: Computational thinking across the curriculum: a conceptual framework. Technical, report college of Computing and Digital Media Technical report(2010)
21. Wang, H.Y., Huang, I., Hwang, G.J.: Effects of an integrated Scratch and project-based learning approach on the learning achievements of gifted students in computer courses. In: 2014 IIAI 3rd International Conference on Advanced Applied Informatics (IIAIAAI), pp. 382–387. IEEE (2014)
22. Wing, J.M.: Computational thinking. Commun. ACM **49**(3), 33–35 (2006)
23. Zavala, L.A., Gallardo, S.C.H., García-Ruíz, M.Á.: Designing interactive activities within Scratch 2.0 for improving abilities to identify numerical sequences. In: Proceedings of the 12th International Conference on Interaction Design and Children. pp. 423–426. ACM (2013)