Available online at www.jlls.org

JOURNAL OF LANGUAGE AND LINGUISTIC STUDIES

ISSN: 1305-578X

Journal of Language and Linguistic Studies, 18(2), 1110-1122; 2022

Computational Thinking And Development Of Cognitive Skills Mediated By Unplugged Activities

César Augusto Hernández Suárez[1], Audin Aloiso Gamboa Suárez[2], Raúl Prada Núñez[3]

[1] Master in Mathematics Education. Research professor at the Universidad Francisco de Paula Santander. E-mail: cesaraugusto@ufps.edu.co. Orcid: https://orcid.org/0000-0002-7974-5560

[2] Doctor in Educational Sciences. Research professor at the Universidad Francisco de Paula Santander. E-mail: audingamboa@ufps.edu.co. Orcid: https://orcid.org/0000-0001-9755-6408

[3] Master in Educational Sciences. Research professor at the Universidad Francisco de Paula Santander. E-mail: raulprada@ufps.edu.co. Orcid: https://orcid.org/0000-0001-6145-1786

APA Citation:

Suárez, C.A.H., Suárez, A.A.G., Núñez R.P., (2022). Computational Thinking And Development Of Cognitive Skills Mediated By Unplugged Activities , *Journal of Language and Linguistic Studies, 18*(2), 1110-1122; 2022.
Submission Date: 12/11/2021
Acceptance Date: 11/02/2022

ABSTRACT

This article shows computational thinking in a high school student's class using a strategy based on unplugged activities to develop and strengthen it. The quantitative method was used with a quasi-experimental approach, through a pretest-posttest design and an intervention with unplugged activities to assess computational thinking skills (algorithmic thinking, decomposition, generalization, abstraction and evaluation). The results indicate that the implementation of the unplugged activities allowed a slight improvement in the development of the students' computational thinking, comparing the pretest and post-test results, especially in the skills of algorithmic thinking, decomposition, and evaluation, but these were not outstanding. It is concluded that it is necessary to conduct a broader study that includes connected and disconnected activities and transversal to other areas.

Keywords: Skills, Computational thinking, secondary education.

INTRODUCTION

The changes accelerated by technological evolution require the development of ICT-related skills, necessary in any field (Chaparro et al., 2018). This brings as a challenge, the need for qualified professionals to meet the challenges of the information society (Zapata-Ros, 2015; González et al., 2017; Aguilar Barreto, 2018), related to product development, technology, innovation, education, skills, employment and employability (Rincón et al., 2022; Salazar-Xirinachs, 2016; Chacón-Guerrero et al., 2017). To achieve this, since the late 1990s and the beginning of the millennium, emphasis has been placed on the inclusion of technology in the curriculum (Sánchez, 2002), and its dimensions to develop the potential of ICT in the classroom, through various uses (Parra & Pincheira, 2011; Urueña, 2016; Hernández et al., 2019; Peret, 2019; Ortiz et al., 2019; Adoumieh, 2021). Some of these uses have emphasized, on the one hand, programming, using visual or non-visual languages and on the other are initiatives based on the idea of computational thinking (García-Peñalvo, 2018; Rodríguez, 2018).

The teaching of programming, as an approach to developing thinking skills in students, emerged in the 1990s from the use of the Logo language and the theory of constructionism (Papert, 1980; 1993; Papert & Harel, 1991a; 1991b), and in more recent times, proposals focused on the cognitive and thinking processes that favor computational and problem-solving competences, such as computational thinking (Wing, 2006, 2008; Resnick et al., 2009; Grover & Pea, 2013; Zapata-Ros, 2015; Denning, 2017), allow, beyond programming.

On the other hand, there is research that evidences the low performance of students in programming classes (Pérez & Pedroza, 2018; Contreras et al., 2019), because they evidence difficulties that are typical of learning programming (Fuentes-Rosado, & Moo-Medina, 2017; Pérez, 2019), so the need arises to develop proposals that go beyond programming, and that allows understanding and solving problems with and without technology, such as those focused on computational thinking and its relationship with problem-solving, its development and application in class (Bell et al., 2009; Csizmadia et al., 2015; Folk et al., 2015; Atmatzidou & Demetriadis, 2016; Lockwood, 2019; Delal & Oner, 2020) through connected (Hernández-Suarez et al., 2022), unplugged (Brennan & Resnick, 2012; García-Peñalvo, 2018; Zapata-Ros, 2019) and mixed (British Council & Ministerio TIC, 2019).

In addition, it is necessary to study the development of computational thinking in students, which is related to thought processes such as abstraction, problem fractionation, generalization of solutions, and pattern analysis among others, as they help to increase their capacity in problem-solving, efficiency and assertiveness in decision making (Rondón, 2020; Hernández-Suárez et al., 2017) which allows them to acquire computational competencies (Millians, 2011) to improve their performance in activities related to technology and computer science, (Wing, 2008; Selby & Woollard, 2013; Palma & Sarmiento, 2015; Valverde et al., 2015; Prada et al., 2019), STEM (Moon et al., 2020), mathematics and science (Weintrop et al., 2016; Hernández-Suárez et al., 2017; Sáez-López et al., 2019; Cabra & Ramírez, 2022) among others.

In this sense, this study shows the results obtained from the analysis of an intervention based on unplugged activities, where the computational thinking skills of middle school students are evaluated. Accordingly, the objective of the study was to diagnose the initial stage of computational thinking skills in a course of high school students and their development once the strategy based on unplugged activities was implemented.

Computational Thinking.

The approach to the development of computational thinking is an emerging field of research, but there is still no consensus on its definition (Brennan & Resnick, 2012). They range from a broader definition posed by Wing (2006, 2008) around problem-solving in diverse areas, where he defines it as a set of thinking skills that combine to solve problems involving mathematics and engineering, which may or may not be solved by a computer to those focused solely on computer science, such as Denning (2017), who defines it as the thinking processes involved in solving problems such that their solutions can be represented as steps and algorithms within a given computational model. Furthermore, according to other authors, computational thinking is used in programming, but should not be limited to it (García-Peñalvo, 2018; Rojas-López & García-Peñalvo, 2018; Román-González et al., 2015; Seoane, 2018).

Therefore, in this study it will be understood, that computational thinking according to Selby & Woollard (2013) and Csizmadia et al. (2015) consists of a series of thinking skills, very much in connection with Bloom's taxonomy (Selby, 2015), such as algorithmic thinking (creating a series of ordered steps to solve a problem), decomposition (breaking problems into subproblems or parts), generalization (analyzing data and looking for patterns), abstraction (eliminating unnecessary details and choosing from good representations) and evaluation (ensuring that a solution, whether an algorithm, system or process, is good), which facilitates complex problem-solving.

Connected vs. Disconnected Computational Thinking

Connected computational thinking is based on the teaching of programming and the development of skills when students program. Within this line are Resnick et al. (2009), based on the ideas of Papert (1980), with their programming language Scratch, developed to bring programming closer to children so that they become involved in the development of digital content and are not only consumers of technology. And, on the other hand, some initiatives promote the development of unplugged computational thinking (without the need for computers), focused on the development of activities and problem solving that are applicable in everyday situations (Bell et al., 2009; Brennan & Resnick, 2012; Folk et al., 2015; García-Peñalvo, 2018; Zapata-Ros, 2019; Hernández et al., 2022) that are far away from a technological device but teach computing concepts in a funny way.

Finally, to assess Computational Thinking, rather than programming concepts, several tests are used, among which the Bebras test stands out (Dagiene, 2006; Dagiene, & Futschek, 2008), which does not require prior knowledge of programming or computer science, and aims to assess computational thinking in students between the ages of 6 and 18 (Lockwood & Mooney, 2018). Each problem is designed for several age ranges: Kits (6-8), Castors (8-10), Juniors (10-12), Intermediates (12-14), Seniors (14-16), Elite (16-18); within each group a degree of difficulty is indicated with A being the lowest difficulty and C being the highest (Bebras UK, 2018).

METHOD

Research method

The study was quantitative, with a quasi-experimental design, because it was applied to an already formed group whose composition cannot be altered, through the application of a pretest-posttest design (Hernandez et al., 2014), to diagnose the initial state of computational thinking skills and their development, once a strategy based on unplugged activities was implemented in a group of high school students (Gómez et al., 2022)

Population and sample

The study population corresponded to students enrolled in a primary and secondary educational institution belonging to the Secretary of Education of the department of Norte de Santander, Colombia. For the sample, middle school students (11th grade) were chosen, since at this level students must handle programming concepts, especially the Scratch language. For the application of the pretest and posttest, 48 students participated, 27 male and 21 female, with an average age of 16.7 years and a standard deviation of 0.9.

Procedure and data collection

The study was carried out in the Technology and Information Technology classroom of the educational institution. A total of 16 sessions were carried out, each lasting 2 hours and two sessions per week, distributed as follows: One for the application of the pretest, 14 for the implementation and development of the unplugged activities, and finally, one for the execution of the posttest, all this process during the first semester of the year 2022 during the school year. The pretest was applied in January, and between February and May the intervention was carried out, and finally, the posttest was applied at the beginning of June.

Instrument

The instrument for assessing computational thinking skills was taken from Lockwood & Mooney's (2018) assessment test with problems released from the Bebras test of the UK annual competitions (Bebras UK, 2016). The Lockwood & Mooney (2018) test has an ascending degree of difficulty throughout the problems with a consistent validation process and structure (Lockwood, 2019).
From the 13 questions in Lockwood & Mooney's (2018), Test 1, 8 problems were selected covering all skills ordered by difficulty in ascending order as each problem progressed, commensurate with the age of the participating students. Each problem is assigned several computational thinking skills. Table 1 lists the selected problems, the level and the skills involved in the problems, according to Selby & Woollard (2013) and Csizmadia et al. (2015).

Table 1. Selected problems, level and associated computational thinking skills.

		Associated computational thinking skills				
Problem	Level	Algorithmic thinking	Decomposition	Generalization	Abstraction	Evaluation
P1. Bracelet	Intermediate A					x
P2. Animation	Intermediate A		x	x	x	x
P3. Cross-country	Intermediate A	x				x
P4. Roll the dice	Intermediate A	x				x

P5. Drawing Stars	Senior B	x	x	x	x	x
P6. Beaver Lunch	Senior B		x	x	x	x
P7. Animal Competitio n	Elite A		x	x	x	x
P8. Computer Stack	Elite B	x	x		x	x

In this context, this instrument with Bebras Problems to assess computational thinking skills is adequate to measure these skills before (pretest) and after (posttest) the intervention. The time allotted to solve the test was one hour of class time, using Google Forms, in the classroom.

The Bebras challenge scores the difficulty levels as follows: for Level A, 6 points if the answer is correct, and -2 if the answer is incorrect; for Level B, 9 points if the answer is correct, and -3 if the answer is incorrect; and for Level C, 12 points if the answer is correct, and -4 if the answer is incorrect. Unanswered questions were assigned a value of zero (0) points. In the case of the tests applied, the minimum score was -19 and the maximum was 57 points.

Intervention through Unplugged activities.

The lessons for the intervention strategy were taken from Bordignon & Iglesias (2020), which are oriented to develop concepts and skills related to computational thinking, based on lessons from Bebras. Each activity is designed to be worked collaboratively, in groups of no more than 4 members. Each lesson is presented in the form of homework and is solved during class time. It is made up of 4 sections: (1) presentation of the task, (2) answer and explanation of the task, (3) to know more (reflections and additional information for teachers and students) and (4) challenges: to continue deepening in the resolution of similar tasks. They are organized sequentially, taking into account what has been seen in previous lessons, increasing their level of complexity from one lesson to another. The tasks implemented are shown in the following Table 2.

Table 2. Tasks developed and applied as an intervention.

	Computational thinking skills				
	Algorithmic thinking	Decomposition	Generalization	Abstraction	Evaluation
W1. Which way does it go?	x	x			
W2. On your way home	x	x			
W3. Escape	x		x		

W4. Editing news				x	x
W5. Animal figures				x	
W6. Magic bracelet		x	x		
W7. Log Art			x	x	
W8. Scrambled Results	x				x
W9. Magic potions	x				x
W10. Party Guests		x			x
W11. Loading the jars		x			x
W12 Secret Agents			x		x
W13 The Wall Painter	, ,		x	x	x
W14 Walking Trees	x		x	x	

RESULTS

General results of the application of the pretest and posttest of computational thinking skills. Table 3 below shows the descriptive statistics corresponding to the pretest and posttest.

Table 3. Descriptive statistics on total score in pretest, posttest and difference of pretest and posttest values.

Statistician	Pretest values	Post-test values	Pretest and posttest difference
Mean	6.4	23.8	17.4
Median	3	24	21
Mode	3	24	21
Standard Deviation	10.5	10.3	-0.2
Variance	110.4	105.7	-4.7
Minimum	-10	3	13
Maximum	24	41	17
Quantile Q1	0	17	17
Quantile Q3	14	31	17

The results presented in Table 3 show an increase in the posttest values. After the implementation of the intervention, the results showed an increase in correct answers (μ = 17.4) concerning the values obtained in the pretest, where arithmetic mean of 23.8 points out of 57 possible points was recorded. However, the standard deviation presents closer results (σ = -0.2), which indicates that the data have a similar dispersion in the application of both tests. The standard deviation in the post-test (σ = 10.3) reflects that there are scores distant from the mean obtained, an example of this is the participants who increased their pretest score by (17) units and also those who reduced it by (13) points.

Table 4 presents a comparison of each group of responses before and after the implementation of the intervention, discriminated by the problem.

.

Table 4. Number and percentage of correct answers per problem.

Problem	Correct answers Pretest	pretest % correct	Correct answers Posttest	Hit % Posttest	Difference between correct answers Posttest - Pretest	Difference % correct Posttest - Pretest
P1. Bracelet	39	81,3%	45	93,8%	6	12,5%
P2. Animation	32	66,7%	38	79,2%	6	12,5%
P3. Cross-country	6	12,5%	25	52,1%	19	39,6%
P4. Roll the dice	20	41,7%	39	81,3%	19	39,6%
P5. Drawing Stars	19	39,6%	26	54,2%	7	14,6%
P6. Beaver Lunch	13	27,1%	17	35,4%	4	8,3%
P7. Animal Competition	4	8,3%	16	33,3%	12	25,0%
P8. Computer Stack	0	0,0%	7	14,6%	7	14,6%

According to the results in Table 4, it is found that students answered correctly 55.5% of the total number of problems in the posttest, with a mode of 4 correct answers per student, compared to the pretest where only 34.7% answered correctly with a mode of 3 correct answers.

On the other hand, the following problems stand out: P3 and P4, which presented in the posttest a significant increase in the number of participants with 39.6% of correct answers; these problems have in common relation with the skills of algorithmic thinking and evaluation. Followed by problems P7, with an increase of 25.0% of hits and problems P5 and P8, with a 14.6% increase in the number of participants, which have in common the relationship with the skills of decomposition and evaluation.

Consequently, after the implementation of the intervention, the problems that support the development of the skills of Algorithmic Thinking, Decomposition and Evaluation presented an increase in the number of subjects with successes concerning those same tasks in the pretest. Concerning problems P1, P2, and P6, the differences between the posttest and pretest are minimal, indicating that the students have difficulties related especially to the abstraction skill.

DISCUSSION

The research analyzed the development of computational thinking skills, using a strategy based on unplugged activities for a course of high school students. A pretest was used to diagnose the initial state of the skills, and a posttest was used to evaluate their development once the intervention was applied. Although the concept of computational thinking is very broad, the analysis was approached from the mental process that comprises the skills proposed by Selby & Woollard (2013) and Csizmadia et al. (2015): algorithmic thinking, decomposition, generalization, abstraction, and evaluation, all parts of computational thinking.

The results show that there is a slight improvement in the development of computational thinking skills, due to the intervention of unplugged lessons, whose effect is based on the mental processes of algorithmic thinking, decomposition, and evaluation. After the intervention, students recognized, interpreted and applied these mental processes to understand what the problem is and thus develop efficient solutions. They applied algorithmic thinking skills to plan and organize instructions, decomposition skills to break it down and understand it, and evaluation to ensure that the solution was useful for its purpose, as proposed by Ortega-Ruipérez (2020). These results corroborate the findings of other studies related to the use of unplugged activities to develop computational thinking (Bell et al., 2009; Delal & Oner, 2020; Folk et al. (2015).

In general, the results were not outstanding in the whole group since there were students who did not reach the same level of development of computational thinking, possibly because they found difficulties when solving problems involving skills such as generalization in the search for a connection between data and identifying patterns, which complicated the search for a solution. Likewise, abstraction was not noticeable; consequently, it is more complicated to generate a representation of the problem they are trying to solve because they do not identify which details are important and which can be omitted (Ortega-Ruipérez, 2020).

However, the need to strengthen computational thinking in the teaching-learning processes in the area of Technology and Computer Science is affirmed, beyond programming, which is a fundamental skill in problem-solving, as stated by authors such as Denning (2017) and Wing (2006, 2008). In addition, the use of unplugged lessons strengthens students' computational thinking and addresses computer science concepts, regardless of whether or not technology is present (Zapata-Ros, 2019). It was found that, with proper planning, it is an alternative to develop these skills, in addition to providing a form of the formative evaluation of them with and without the use of ICT, following the proposal of Bebras' problems (Dagiene & Futschek, 2008; Lockwood & Mooney, 2018). Such a strategy was fundamental to promoting computational thinking and contributes information to the ongoing debate, regarding the adequacy of technology and computer science curricula (Said, 2015; Hernández et al., 2019), and which becomes more necessary, after the educational emergency caused by the Covid-19 (Prada et al., 2022).

CONCLUSIONS

The overall results obtained showed an increase in the development of computational thinking after the intervention with unplugged activities. It should be noted that the concept of computational thinking is still very broad. In particular, this study explored a way to evaluate the effects on the development of computational thinking by approaching the phenomenon from the cognitive process that comprises the coverage of the five skills such as algorithmic thinking, decomposition, generalization, abstraction and evaluation.

On the other hand, the research showed some methodological limitations such as the small size of the sample, the lack of an instrument to assess the students' perception of the strategy implemented, the use of mixed activities (connected and unplugged), complementing the quantitative results with other qualitative methods, including other contexts and educational actors, among others, as future studies to perform a more complete analysis and obtain a broader view of computational thinking in the educational context.

REFERENCES

Adoumieh Coconas, N. (2021). La didáctica de la lengua mediada por las TIC Storyjumper como propuesta innovadora en la creación de cuentos. Revista Perspectivas, 6(1), 101–113. https://doi.org/10.22463/25909215.2928

Aguilar Barreto, A. J. (2018). Servicio educativo: una revisión legislativa de su estructuración frente a las obligaciones del estado. Revista Perspectivas, 3(2), 110–120. https://doi.org/10.22463/25909215.1594

Atmatzidou, S., & Demetriadis, S. (2016). Advancing students' computational thinking skills through educational robotics: A study on age and gender relevant differences. Robotics and Autonomous Systems, 75, 661–670. https://doi.org/10.1016/j.robot.2015.10.008

Bebras UK. (2016). Have a go at past competitions. http://beaver-comp.org.uk/

Bebras UK. (2018). UK Bebras. http://www.bebras.uk/

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer Science Unplugged: school students doing real computing without computers. The New Zealand Journal of Applied Computing and Information Technology, 13(1), 20-29.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. American Educational Research Association, 1–25.

British Council & Ministerio de Tecnologías de la Información y Comunicaciones. (2019). Guía para la enseñanza y el aprendizaje del pensamiento computacional Pensamiento computacional.https://aulatic21.files.wordpress.com/2019/05/guc3ada-para-la-ensec3b1anza-y-el-aprendizaje-del-pensamiento-computacional.pdf

Bordignon, F., & Iglesias, A. A. (2020). Introducción al pensamiento computacional. UNIPE, Editorial Universitaria, Educar S.E.

Cabra, M. L., & Ramírez, S. A. (2022). Desarrollo del pensamiento computacional y las competencias matemáticas en análisis y solución de problemas: una experiencia de aprendizaje con Scratch en la plataforma Moodle. Revista Educación, 46(1). https://doi.org/10.15517/REVEDU.V46I1.44970

Chacón-Guerrero, E., Eslava-Zapata, R., Chacón-Lobo, G., González-Júnior, H. A., & Buitrago-Rodríguez, M. del V. (2017). Cooperación académica entre el sector productivo y las universidades: Una evaluación basada en las prácticas estudiantiles. Revista Perspectivas, 2(2), 17–31. https://doi.org/10.22463/25909215.1311

Chaparro, J. C., Jaimes, M. C., & Prada, R. (2018). Modernidad y Educación: Una reflexión a propósito de sus legados y desafíos. Revista Perspectivas, 3(2), 121–132. https://doi.org/10.22463/25909215.1672

Contreras, J. V., Ramírez, X., & Hernándezl, V. K. (2019). Factores que influyen en el desempeño escolar de los estudiantes de Básica Primaria de una institución educativa del área metropolitana de Cúcuta. Revista Perspectivas, 4(1), 6–13. https://doi.org/10.22463/25909215.1746

Csizmadia, A., Curzon, P., Dorling, M., Humphreys, S., & Ng, T. (2015). Computational Thinking - A guide for teachers. Computing At School.

Dagienė, V. (2006). Information technology contests – introduction to computer science in an attractive way. Informatics in Education, 5(1), 37-46.

Dagiene, V., Futschek, G. (2008). Bebras International Contest on Informatics and Computer Literacy: Criteria for Good Tasks. In: R. T. Mittermeir, M. M. Syslo (Eds.), Informatics Education – Supporting Computational Thinking. (Vol. 5090) (pp. 19-30). Springer.

Delal, H., & Oner, D. (2020). Developing middle school students' computational thinking skills using unplugged computing activities. Informatics in Education, 19(1), 1–13. https://doi.org/10.15388/infedu.2020.01

Denning, P. J. (2017). Viewpoint Remaining Trouble Spots with Computational Thinking. Communications of the ACM, 60(6), 33–39. https://doi.org/10.1145/2998438

Folk, R., Lee, G., Michalenko, A., Peel, A., & Pontelli, E. (2015). GK-12 DISSECT: Incorporating Computational Thinking with K-12 Science without Computer Access. 2015 IEEE Frontiers in Education Conference (FIE). 1-8. https://doi.org/10.1109/FIE.2015.7344238

Fuentes-Rosado, J. I., & Moo-Medina, M. (2017). Dificultades de aprender a programar. Revista Educación en Ingeniería, 12(24), 76-82. https://doi.org/10.26507/rei.v12n24.728

García-Peñalvo, F. J. (2018). Pensamiento Computacional. IEEE Revista Iberoamericana de Tecnologías del Aprendizaje. 13(1), 17-19. https://doi.org/10.1109/RITA.2018.2809939

González, M. V., Martínez, M. A., De Zayas, M. R., & López, J. (2017). Nuevos espacios y retos para los actuales profesionales de la información. Revista Interamericana De Bibliotecología, 40(3), 317-325. https://doi.org/10.17533/udea.rib.v40n3a10

Grover, S., & Pea, R. (2013). Computational Thinking in K–12: A Review of the State of the Field. Educational Researcher, 42(1), 38–43. https://doi.org/10.3102/0013189X12463051

Vanegas, J., Gamboa, A., & Goméz, J. (2022). Perspectivas epistemológicas en el locus de las Ciencias Cognitivas. Jotamar.

Hernández, R., Fernández, C., & Baptista, P. (2014). Metodología de la investigación (6ª. Ed.). McGraw-Hill.

Hernández, C. A., Gamboa, A. A., & Avendaño, W. R. (2022). Diseño de algoritmos en tecnología con Scratch para el desarrollo del Pensamiento Computacional. Revista Boletín Redipe, 11(2), 461–476. https://doi.org/10.36260/rbr.v11i2.1696

Hernández-Suárez, C. A., Gamboa-Suárez, A. A., & Avendaño-Castro, W. R. (2022). Design Of Algorithms For Developing Computational Thinking In Elementary And Middle School Students. Journal of Positive Psychology and Wellbeing, 6(2), 90-107.

Hernández-Suárez, C.A., Prada-Núñez, R., & Gamboa-Suárez, A.A. (2017). Conocimiento y uso del lenguaje matemático en la formación inicial de docentes en matemáticas. Revista de Investigación, Desarrollo e Innovación, 7(2), 287–299. https://doi.org/10.19053/20278306.v7.n2.2017.6071

Hernández-Suárez, C.A., Prada-Núñez, R., & Ramírez-Leal, P. (2017). Obstáculos epistemológicos sobre los conceptos de límite y continuidad en cursos de cálculo diferencial en programas de ingeniería. Revista Perspectivas, 2(2), 73–83. https://doi.org/10.22463/25909215.1316

Hernández, C, A., Prada, R., & Ramírez, P. (2019). Competencias TIC e investigativas entre docentes de educación básica. Ecoe Ediciones.

Lockwood, J. (2019). Computer Science To Go (CS2Go): Developing a course to introduce and teach Computer Science and Computational Thinking to secondary school students (master's thesis, National University of Ireland Maynooth). Institutional Repository Maynoothuniversity. http://mural.maynoothuniversity.ie/11015/

Lockwood, J., & Mooney, A. (2018). Developing a Computational Thinking Test using Bebras problems. CC-TEL/TACKLE@EC-TEL. http://mural.maynoothuniversity.ie/10316/

Millians, M. (2011). Computational Skills. In: S. Goldstein & J. A. Naglieri (eds.) Encyclopedia of Child Behavior and Development. Springer. https://doi.org/10.1007/978-0-387-79061-9_645

Moon, J., Do, J., Lee, D., & Choi, G. W. (2020). A conceptual framework for teaching computational thinking in personalized OERs. Smart Learning Environments, 7(6). https://doi.org/10.1186/s40561-019-0108-z

Ortega-Ruipérez, B. (2020). Pedagogía del Pensamiento Computacional desde la Psicología: un Pensamiento para Resolver Problemas. Cuestiones Pedagógicas. Revista de Ciencias de la Educación, 2(29), 130-144. https://doi.org/10.12795/CP.2020.i29.v2.10

Ortíz-Arismendy, A., Ramírez, M. E., & Diaz, M. E. (2019). El portafolio digital como herramienta de apoyo en el proceso de enseñanza. Revista Perspectivas, 4(2), 20_30. https://doi.org/10.22463/25909215.1967

Palma, C. A., & Sarmiento, R. E. (2015). Estado del arte sobre experiencias de enseñanza de programación a niños y jóvenes para el mejoramiento de las competencias matemáticas en primaria. Revista Mexicana de Investigación Educativa, 20(65), 607–641.

Papert, S. (1980). Mindstorms: children, computers, and powerful ideas. Basic Books.

Papert, S. (1993). The Children's Machine: Rethinking School in the Age of the Computer. Basic Books.

Papert, S. & Harel, I. (1991a). Constructionism. Ablex Publishing

Papert, S. & Harel, I. (1991b). Situating Constructionism. In S. Papert & I. Harel (ed.), Constructionism. Ablex Publishing.

Parra, E., & Pincheira, R. (2011). Integración curricular de las TIC. Quaderns digitals: Revista de Nuevas Tecnologías y Sociedad, (68).

Peret, M. (2019, 02 de septiembre). La inclusión curricular de las tecnologías en los procesos de enseñanza y de aprendizaje. https://maximilianoperet.com.ar/wp/2019/09/02/la-inclusion-curricular-de-las-tecnologias-en-los-procesos-de-ensenanza-y-aprendizaje/

Pérez, J. A., & Pedroza, O. V. (2018). LM1: una metodología de estudio para la asignatura "Programación 1". Educere: Revista Venezolana de Educación, (73), 635-648.

Pérez, J. A. (2019). El pensamiento computacional en la vida cotidiana. Revista Scientific, 4(13), 293-306. https://doi.org/10.29394/Scientific.issn.2542-2987.2019.4.13.15.293-306

Prada, R., Hernández, C. A., & Gamboa, A. A. (2019). Different scenarios for the teaching of mathematics with the support of virtual platforms: Flipped classroom. Journal of Physics: Conference Series, 1388(1), 012046:1-5.

Prada, R., Hernández, C, A., & Avendaño, W. (2022). Educación y Pandemia: Afectaciones del Covid-19 en Actores Educativos. Ediciones Nueva Jurídica.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B., & Kafai, J. (2009). Scratch: Programming for All. Communications of the ACM, 52(11), 60–67. https://doi.org/10.1145/1592761.1592779

Rincón, I. B., Rengifo, R., Hernández, C., & Prada, R. (2022). Educación, innovación, emprendimiento, crecimiento y desarrollo en América Latina. Revista De Ciencias Sociales, 28(3), 110-128. https://doi.org/10.31876/rcs.v28i3.38454

Rodríguez, R. E. (2018). La realidad virtual en la migración de plataformas educativas de 2D a 3D en los escenarios de las Universidades de San José de Cúcuta. Revista Perspectivas, 3(2), 86–96. https://doi.org/10.22463/25909215.1592

Rojas-López, A., & García-Peñalvo, F. J. (2018). Escenarios de Aprendizaje para la Asignatura Metodología de la Programación a partir de Evaluar el Pensamiento Computacional de. 6. VAEP-RITA, 6(1), 15-22.

Román-González, M., Pérez-González, J. C., & Jiménez-fernández, C. (2015, 14-16 de octubre). Test de Pensamiento Computacional : diseño y psicometría general Computational Thinking Test : design & general psychometry. III Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC). https://doi.org/10.13140/RG.2.1.3056.5521

Rondón, G. A. (2020). Propuesta para desarrollar habilidades de pensamiento computacional en estudiantes de décimo grado del Colegio Facundo Navas Mantilla (tesis de maestría, Universidad Autónoma de Bucaramanga). Repositorio Institucional UNAB. https://repository.unab.edu.co/bitstream/handle/20.500.12749/11689/2020_Tesis_Gabriel_Andres_Rondon_Barragan.pdf?sequence=1&isAllowed=y

Sáez-López, J. M., Sevillano-García, M. L., & Vazquez-Cano, E. (2019). The effect of programming on primary school students' mathematical and scientific understanding: educational use of mBot. Educational Technology Research and Development, 67, 1405–1425. https://doi.org/10.1007/s11423-019-09648-5

Salazar-Xirinachs, J. M. (2016). El futuro del trabajo, el empleo y las competencias en América Latina y el Caribe. Pensamiento Iberoamericano, (2), 54-66.

Said, E. (2015). Hacia el fomento de las TIC en el sector educativo en Colombia. Editorial Universidad del Norte.

Sánchez, J. (2002). Integración curricular de las TICs: conceptos e ideas. En: M. J. Fernández, M. Llamas, L. E. Anido (coord.), 6 Congreso Iberoamericano, 4 Simposio Internacional de Informática Educativa, 7 Taller Internacional de Software Educativo. Universidade de Vigo, Servizo de Publicacións. http://www.c5.cl/mici/pag/papers/inegr_curr.pdf

Seoane, A. M. (2018). Pensamiento Computacional entre Filosofía y STEM. Programación de Toma de Decisiones aplicada al Comportamiento de "Máquinas Morales" en Clase de Valores Éticos. VAEP-RITA, 6(1), 4–14.

Selby, C. C. (2015). Relationships: computational thinking, pedagogy of programming, and Bloom's Taxonomy. WiPSCE '15 Proceedings of the Workshop in Primary and Secondary Computing Education, 80-87. https://doi.org/10.1145/2818314.2818315

Selby, C., & Woollard, J. (2013). Computational Thinking: The Developing Definition. University of Southampton. https://eprints.soton.ac.uk/356481/

Urueña, S. (2016). Dimensiones de la inclusión de las TIC en el currículo educativo: una aproximación teórica. Revista Interuniversitaria, 28(1), 209-223. https://doi.org/10.14201/teoredu2016281209223

Valverde, J., Fernández, M. R., & Garrido, M. (2015). El pensamiento computacional y las nuevas ecologías del aprendizaje. RED-Revista de Educación a Distancia, 46(3), 1-18. https://doi.org/10.6018/red/46/3

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Science Classrooms. Journal of Science Education and Technology, 25(1), 127–147. https://doi.org/10.1007/s10956-015-9581-5

Wing, J. (2006). Computational Thinking. Communications of the ACM, 49(3), 33-35. https://doi.org/10.1145/1118178.1118215

Wing, J. (2008). Computational thinking and thinking about computing. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 366, 3717–3725. https://doi.org/10.1098/rsta.2008.0118

Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. Revista de Educación a Distancia (RED), 46(4). https://doi.org/10.6018/red/46/4

Zapata-Ros, M. (2019). Pensamiento computacional desenchufado. Education in the Knowledge Society (EKS), 20, 18-29. https://doi.org/10.14201/eks2019_20_a18