Analysis of the Teaching of Programming and Evaluation of Computational Thinking in Recently Admitted Students at a Public University in the Andean Region of Peru

Ronald Paucar-Curasma[1], Ismar Frango Silveira[2], David Rondon[3], Zenón Manuel López Robles[4], Luís Enrique Porras-Ccancce[5]

[1] *Universidad Nacional Autónoma de Tayacaja Daniel Hernández Morillo, Jr. Bolognesi N° 416, Pampas, 09000, Perú*
[2] *Universidade Presbiteriana Mackenzie, Rua da Consolação 930, São Paulo, 01302-907, Brasil*
[3] *Universidad Continental, Calle Alfonso Ugarte 607, Arequipa, 04001, Perú*
[4] *Universidad Nacional Daniel Alcides Carrión, Pasaje 28 de Julio – Yanacancha, Cerro de Pasco, 19001, Perú*
[5] *Universidad Nacional de Huancavelica, Paturpampa S/N, Huancavelica, 09001, Perú*

Summary

This article analyzes the teaching of programming and computational thinking in students recently admitted at a public university in the Andes of Peru. For this, a survey was carried out on the previous teaching of programming concepts in the schools of origin of the students; also, the computational thinking of these students was evaluated by means of reagents corresponding to the skills of abstraction, decomposition, algorithmic design, generalization and evaluation. In the evaluation of computational thinking, more than 60% of students presented skills of decomposition, abstraction, algorithmic design and evaluation. These results reflect that the students evaluated have competencies in mathematical reasoning exercises that are similar to those required by the reagents used.

Keywords

Computational thinking; programming; reagents; skills.

1. Introduction

Information and communication technologies (ICT) are the basic technologies for the multiple activities of contemporary and future society. In the current social context, the development of education, like other sectors of goods and services, is closely related to the development of technology [1] – a situation that has been revealed and materialized unexpectedly in the recent COVID-19 pandemic. Today, it is not enough to use technology as a support to contribute to the achievement of effective results in the teaching and learning process [2]; but to evaluate ourselves if we are contributing to the training of competent and innovative professionals with the appropriate methodologies and tools. It is known that, in the current scenario of science and technology, it is necessary to have more professionals in STEAM; therefore, educational institutions must contribute to the training of citizens who contribute with creativity and innovation in the digital field. In that sense, developing computational thinking (CT) skills – such as those defined in Wing's initial work [3], which includes the ability to build algorithmic solutions and implement them in a programming language, contributes to the formation of individuals for the twenty-first century and helps in reducing the digital gaps that leave traces throughout Latin America. To deal with this situation, in the school stage, to the teaching

CITIE 2022: Congreso Internacional de Tecnología e Innovación Educativa, November 08-10, 2022, Arequipa, Peru
EMAIL: rpaucarc@unat.edu.pe (R. Paucar-Curasma); ismar.silveira@mackenzie.br (A. 2); drondon@continental.edu.pe (A. 3); ismar.silveira@mackenzie.br; zlopezro@undac.edu.pe (Z. M. Lopéz); luis.porras@unh.edu.pe (L. E. Porras)
0000-0002-9489-576X (R. Paucar-Curasma); 0000-0001-8029-072X (I. F. Selveira); 0000-0003-3506-5309 (D. Rondon); 0000-0003-0744-6075 (Z. M. Lopéz); 0000-0002-2144-9339 (L. E. Porras)

of basic courses in communications, mathematics, arts, etc., we must add courses in digital culture, computational thinking and programming [4].

In the school stage, the courses related to these elements should focus on two aspects: learning computer science and contributing to the development of new computer devices or equipment [5]. But, today, most school-level educational institutions assume a simply utilitarian position in relation to digital technologies, since they limit themselves to teaching office packages, turning students into simple users of computer tools. This position on how to handle computer science teaching, although it may have been necessary, today is insufficient and anachronistic, since in today's world it is necessary to develop skills in mathematical reasoning, logic, algorithmic design and problem solving. Among different approaches, these challenges can be achieved, for example, through the teaching of programming from an early age. Therefore, it is important that students have the necessary cognitive tools to function successfully in the digital world. It is in this sense that an approach based on the principles of computational thinking is proposed, as a set of skills to be developed to solve complex problems [3], [6], [7]

This article analyzes the teaching of programming and evaluates CT in students recently admitted at a public university in the Andes of Peru. To this end, a survey was carried out to know the previous teaching of programming concepts in the schools of origin of the students; also, computational thinking was evaluated using reagents corresponding to the skills of decomposition, algorithmic design, abstraction, evaluation and generalization.

2. Literature review

This section reviews the scientific literature regarding the importance of programming at school age, and computational thinking at school age; as well as programming and computational thinking in the world.

2.1. The importance of school-age programming

Several authors express the importance and benefits of computational thinking in higher education. With regard to computational thinking skills, they highlight abstraction and algorithmic thinking in strengthening competencies in reading comprehension and in solving complex problems following algorithmic methods [8]–[12]; also, [13] in a study carried out, it points out that programming to carry out mathematical tasks and other challenging disciplines can improve the student's understanding, this involves the use of the computer as a tool to solve problems in this area. [14] It points out the use of tools based on block programming and interaction with hardware, generates interest and motivation in students; as well as, teamwork and problem solving through skills of abstraction, decomposition and algorithmic thinking. It also directly relates the development of common skills through creative programming and innovation [15], [16]. Finally, [47][17] it adds critical thinking as a form of reasoning and exchange of ideas prior to solving problems through computational thinking skills. Figure 1 shows the key words that exist about "computational thinking in higher education". The words that stand out the most are: engineering education, mathematical programming, visual languages, robot programming, scratch, visual programming, problem-solving skills, digital literacy, skills and among others.

In our current society, known as the knowledge and information society, current demands give rise to new forms of development of appropriate thinking, so new forms or techniques are needed to help increase knowledge. Years before all the recent changes driven by digital evolution, the seminal work of [18], represent the first reasoning about systems thinking skills as a problem-solving tool, and associated with that the ability to execute computer programming, which together with the ability to design algorithms provide students with skills and abilities to solve problems in general, whether in mathematics, communications, arts and other disciplines at the school stage [19] [48]. The incorporation of programming in school education is, in this way, considered as an important strategy that helps students solve academic and contextual problems. As they state [13] in their study on problem solving related to mathematics, "programming computers to carry out mathematical tasks and other challenging disciplines can improve student understanding; this implies the use of the computer as a tool to solve problems in this area." This statement agrees with what has been pointed out by [20], which emphasizes that the ability to program provides important benefits for the student; for example, it greatly expands the range of what can be created with the computer, it also expands learning through the computational thinking method that is helped by programming to solve complex problems. Thus, also, the same authors affirm that the act of programming computers has the potential to convert students to be innovative and creative, so they will be participants in the creation of technological products that solve real problems of society.

2.2. Computational thinking at school age

There are many reasons and motivations for including CT in regular basic education, among which we can mention: acquisition of a more concrete mastery over technology, a broader and more systemic understanding of the world, identification of transversality elements in different areas, promotion of digital literacy, increased productivity, help in learning other disciplines, promotion of gender equity and inclusive education, reduction in physical limitations and expansion of the possibilities of teamwork, for example. Therefore, in the present twenty-first century, educated citizens are needed to participate and contribute to the development of science and technology; to do this, they must have digital skills and abilities (which include programming). In this sense, educational systems should train students with skills that involve strengthening the CT through hardware and software components; for example, program development, programming of educational robots, process simulation, development of electronic prototypes, among others. In that sense, it is appropriate to generate these skills from an early age.

Two main trends emerge regarding the justification for including CT in regular basic education [21]: the first, the strengthening of CT in students, so that they think like computer science professionals, trained to communicate the problems solved by different means (web, mail, social networks, for example) and analyze/evaluate the problems of their context from a different and real perspective; the second, the dissemination of the benefits of the CT, in the family and national environment.

According to [22] with the integration of the CT, students will have the following competencies: solve problems applying the skills of abstraction, algorithmic design, process automation, data collection and synthesis; testing and debugging developed programs; model, run simulations and reflect/evaluate the results. In addition, students will be aware of the concepts and skills of the CT to apply in different disciplines such as mathematics, social sciences, communication, arts and others.

Since Wing's CT definition (2006), various interpretations or dissertations have proliferated regarding the ecosystem of computational thinking. For example, terms, such as computing, programming or coding and CT: these terms are not identical or the same, but they are part of a large ecosystem. Computing, for most people, is a process evoked when a computational agent (computer or other electronic device) acts on its inputs under the control of an algorithm [23]; while programming or coding is writing code or program (a set of instructions) to be interpreted and executed by a computer or other electronic device; instead the CT applies the skills of abstraction, decomposition, pattern recognition and algorithms in solving complex problems, with the help of a computational agent or computer [24]. In that same context, Wing points out that the CT is to think like a professional of computer science, when a problem is solved; understanding by computer science, the discipline that systematically studies the elaboration of algorithms, description and transformation of information, with theoretical foundation for efficient design and implementation [25]. Figure 1 shows the space between computational thinking, computation, and programming or coding [26].

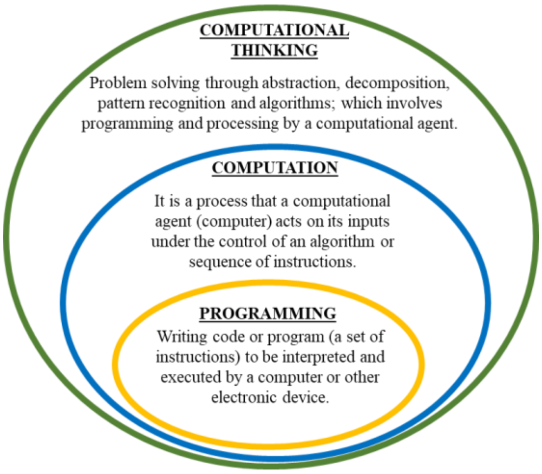


Figure1: Space between computational thinking, computation, and programming

2.3. Programming and computational thinking in the world

In North America, at the beginning of 2016, the "computer science for all" program was launched [27], with the aim of training students from the initial stage to high school in topics related to computer science and CT skills, with the aim that in the very near future they will be creators of the digital economy. Not only to be consumer users, but to be active citizens to generate changes in national and international society. Under this proposal, [28] they point out that computer science in school education (K-12) promotes the teaching of the CT and computer programming; also, it states that the CT "is the human capacity to formulate problems so that their solutions can be represented as computational or algorithmic steps to be carried out by an information processing agent."

In Europe, the Ministry of Education (2018) since 2012, has developed several recommendations on the integration of the CT in the curriculum at school stage, supported by the Royal Society and the ACM (Association of Computing Machinery). Also, the European Commission [30] participated in these proposals, considered politically the point of reference for the integration of the CT in the countries that make it up. [31] states that the reasons for the integration of programming into the curriculum of educational institutions in European countries are intended to generate improvements in twenty-first century skills. For example, in Finland the CT and the development of computer programs were included as a strategy for generating digital skills in students, applying to mathematics and crafts [32]. As part of this process, manuals or instructions were developed to teach CT skills and computer programming; they also provided training for different target groups: principals, teachers and educational authorities.

In Brazil, the [33] and [34] proposed the teaching of computing in the curriculum of regular basic education, under 3 main axes: the CT axis focused on the teaching of abstraction, analysis and automation; the digital world axis focused on the teaching of coding, data processing and representation; finally the digital culture axis focused on technological understanding, digital ethics and computing in society. The proposal was recently approved by the National Council of Education.

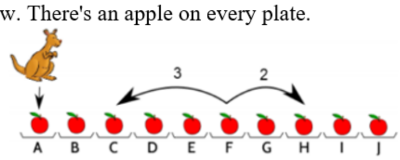
In Peru, there are independent initiatives by universities on CT [35]–[39]; there are still no proposals at the central government level for the teaching or integration of computing into the curriculum of regular basic education.

3. Research methodology

The research was conducted at the Universidad Nacional Autónoma de Tayacaja Daniel Hernández Morillo in the Huancavelica region, located in the Andes of Peru. The recent students belong to the 2021-I period of the industrial engineering career; 37 female and male students participated in the study. To know the teaching of programming according to the school of origin, a survey was applied based on 9 questions, of which 2 questions were selected: "Indicate the origin of the educational institution where you completed your secondary studies" and "With respect to computer teaching in your school, indicate the topic studied"

CT assessment was based on 05 items [40]–[42]; these reagents are related to the skills of decomposition, abstraction, generalization, algorithmic design and evaluation of the CT. Figure 2 shows part of the reagents used.

Exercise 2 (Abstraction)
There are 10 dishes in a row. There's an apple on every plate.



The kangaroo Thomas loves to jump. First, he jumps from the leftmost plate with the letter A. At each jump after the initial, jumps two plates forward, or three plates backward. (An example of the two possible jumps from a plate is shown with arrows in the image.)
Thomas just jumps into plates with an apple.
If he jumps onto a plate, he picks up the apple.
If Thomas picks all 10 apples. Which apple does he pick at the end? A, B, C, D, E, F, G, H, I, or J.

Figure 2: Kangaroo reagent corresponding to abstraction skill

4. Results

Below are the results of the survey conducted on the type of educational institution of origin and the teaching of programming in schools. Thus, the results of the evaluation of the skills of the CT based on the reagents used are also shown.

4.1. Survey results

Regarding the origin of the type of educational institution (public or private), Figure 3 shows that 78% of students come from the public sphere; meanwhile, 22% come from the private sector. In Figure 4, it is shown that 78.4% of students took a computer course and within this course 73% followed office automation packages (Word, Excel, Power point, others); meanwhile, only 2.7% carried programming and 2.7% web concepts. Also, it is observed that 21.6% of the students did not take computing course.

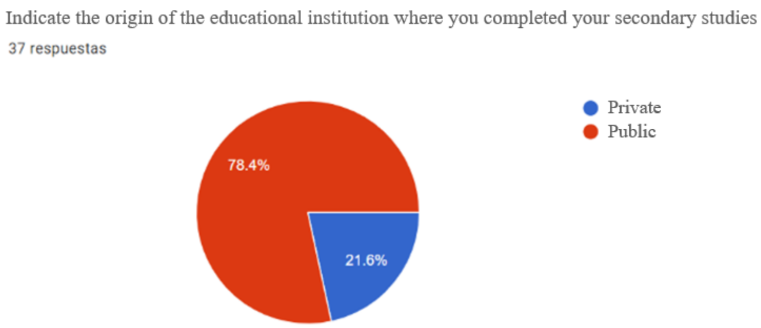


Figure 3: Type of educational institution of origin

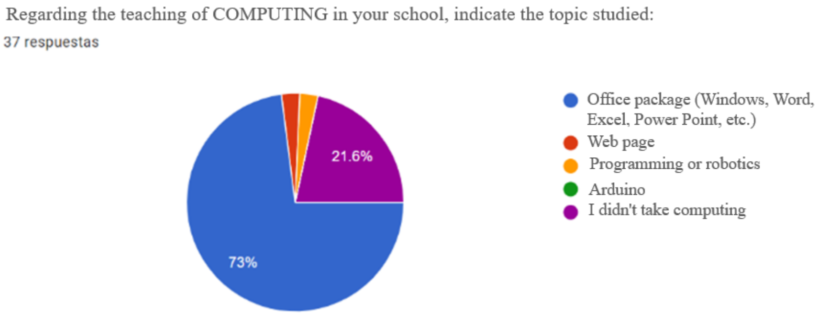


Figure 4: Teaching of the computing course in schools of origin

4.2. Results of the evaluation of computational thinking

Of a total of 37 students recently admitted at university, the CT was evaluated through the reagents corresponding to the skills of decomposition, abstraction, generalization, algorithmic design and evaluation. Figure 5 shows that 62.2% of students correctly answered the Mobile reagent (Decomposition). For the Kangaroo reagent (Abstraction) 62.2% of students answered correctly, as shown in Figure 6. 56.8% of students correctly answered the Spies on the Move (Generalization) reagent, as shown in Figure 7. The Beavers reagent (Algorithmic Design) obtained the highest percentage of correct answers with a value of 73%, as shown in Figure 8. Finally, Figure 9 shows that 75.7% of students correctly answered the Salto de Charco reagent (Evaluation).

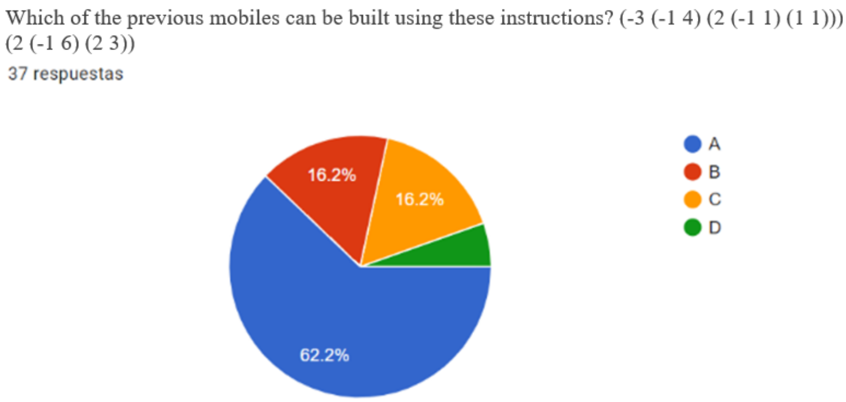


Figure 5: Evaluation of decomposition ability (Mobile reagent)

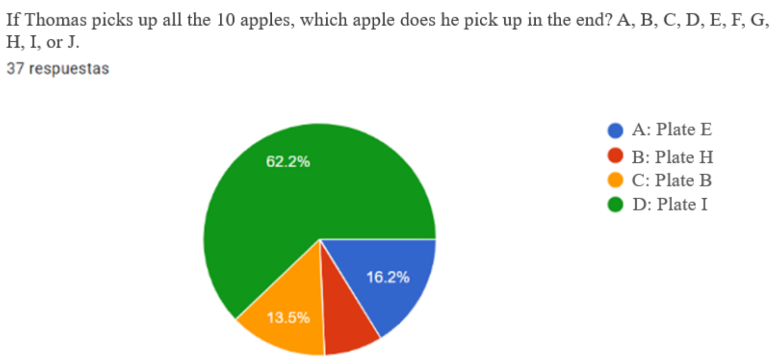


Figure 6: Evaluation of abstraction ability (Kangaroo reagent)

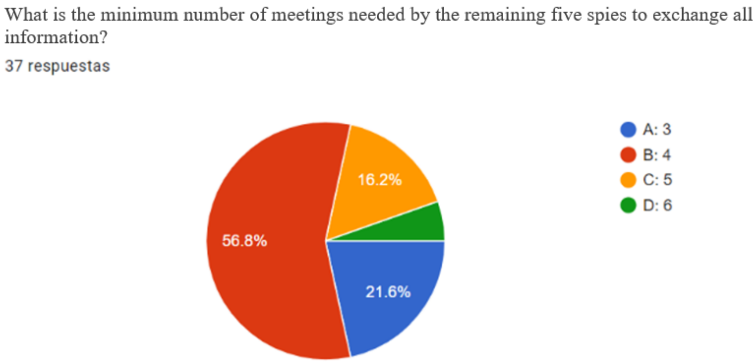


Figure 7: Generalization Skill Assessment (Spies reagent)

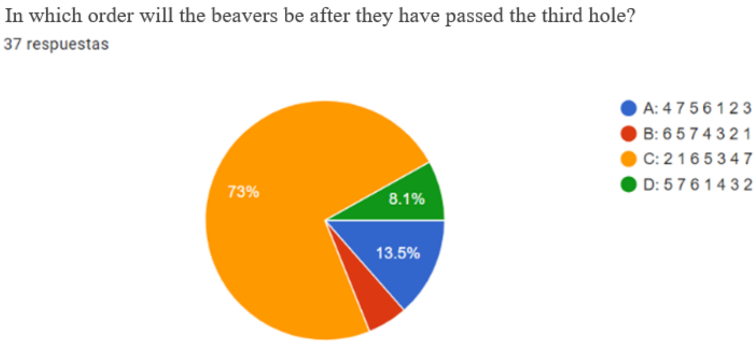


Figure 8: Evaluation of algorithmic design skill (beaver reagent)

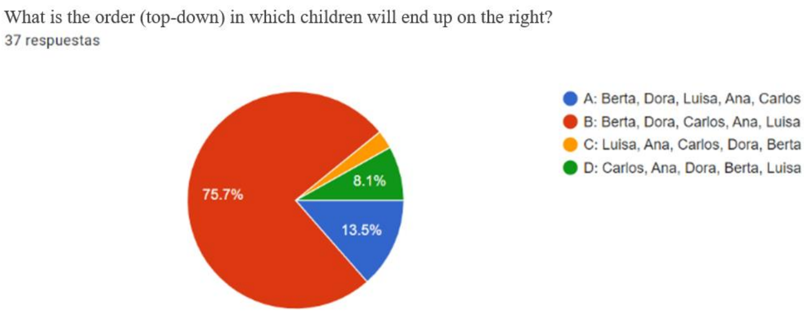


Figure 9: Evaluation of evaluation skill (Salto de Charco reagent)

5. Discussions

Activities or Problem-solving processes are used to identify, analyze, reflect on experiences and ways of conducting research in engineering or STEM-related areas; in this process, the problem, the research methods to be followed, the implementation and evaluation of the solution are presented; that is, problem solving is an inquisitive domain where students constantly ask questions, identify conjectures or relationships, look for various ways to support and implement it and then evaluate and communicate the results [43], [44]. From the results obtained with respect to the teaching of computers in school education at the secondary level, they show that 78.4% of students took the computer course, and within this course 73% followed office packages (Word, Excel, Power point, others), 2.7% followed programming, 2.7% followed web page and 21.6% did not take the computing course. Here, it can be seen that schools mostly only teach office packages; this form of teaching, based on a utilitarian approach, turns students into users of computer tools, but not as innovators or creators of technology [4] [46]. The same is pointed out, when stating that the computer is used to think in the key of today's world. Therefore, it is considered that the teaching of programming is important to develop cognitive processes in students; in this context, the same authors affirm that school education should focus on programming first, and the use of office automation later. In private schools, the insertion of technology is mostly before public schools; this study shows that the majority of students come from the public sphere; normally, these schools are characterized by limited economic resources and therefore do not have trained teachers who could teach programming courses or are not trained because they live in remote areas and rugged geography. It should be noted that the university is located in one of the poorest regions of Peru and most of the students come from that geographical space.

Regarding the assessment of computational thinking skills, the percentage of students who answered correctly the most correspond to the assessment skill, followed by the algorithmic design skill; meanwhile, regarding the ability of abstraction and decomposition, the same number of students answered correctly, and the generalization ability that corresponds to 56.8% was the lowest number of students to answer correctly.

From the results of the evaluation of computational thinking, it is observed that most students have competencies in the skills of evaluation, algorithmic design, abstraction and decomposition, which correspond to 75.7%, 73%, 62.2% and 62.2% of students respectively; these results are related to the nature of the engineering career, which normally works in students with competencies in logic and mathematical reasoning, subjects that are related to the reagents of computational thinking [45].

6. Conclusions

Among the CT skills to be developed in students, this article pointed out the importance of teaching programming for the development of cognitive processes in students. The educational centers within their curricular plan have the computing course, which mostly teach simply office packages. These computer utilities contribute little to the development of computational thinking in students; therefore, it is recommended to teach programming first and then office packages, if necessary. Currently, there are programming tools to train students to write programs: mBlock, Arduino, Lightbot and Scratch, which are characterized by being friendly and intuitive to train students without programming experience. With these programming tools, the student could follow a methodology for learning programming, which comprises problem analysis, algorithmic design, translation of the algorithm into computer code and finally program debugging.

The evaluation of computational thinking in recent college students was carried out satisfactorily according to plan. It was verified that the reagents used to evaluate computational thinking skills are adequate and gives us an indicator of the number of students who have competencies in abstraction, decomposition, algorithmic design, generalization and evaluation; thus, also for students who answered some items incorrectly, educational strategies can be applied to improve computational thinking skills. To teach the university courses related to programming, it is recommended to apply the reagents to know the level of computational thinking skills in students prior to the development of the course; therefore, according to the results obtained from the evaluation, different educational strategies could be applied to reinforce the skills in students during the development of the course; carrying out these processes in the first cycles of the university is fundamental, so that students in later cycles have a positive attitude and confidence in solving complex problems related to the courses and situations of their context.

7. References

[1]A. Toffler, Shocks, Surges, and Power in the Digital Age, READ, Anne. Mexico: Prentice Hall, 2001.

[2]A. J. Berlanga, F. J. G. Peñalvo, y P. B. Sloep, «Towards eLearning 2.0 University», Interact. Learn. Environ., vol. 18, n.o 3, pp. 199-201, 2010, doi: 10.1080/10494820.2010.500498.

[3]J. M. Wing, «Computational Thinking», Commun. ACM, vol. 49, n.o 3, pp. 267-271, 2006.

[4] F. Llorens, "They say out there... that the new literacy passes through programming", Revision, vol. 8, no. 2, pp. 11-14, 2015.

[5]J. L. Sierra-Rodríguez and F. J. García-Peñalvo, "Computers in Education and Education in Computer Science | Educational Informatics and Education in Computer Science", Theory. la Educ., vol. 16, no. 4, pp. 25-31, 2015, doi: 10.14201/eks20151642531.

[6]F. Llorens, F. J. García-Peñalvo, X. Molero, and E. Vendrell, «La enseñanza de la informática, la programación y el pensamiento computacional en los estudios preuniversitarios [The Teaching of

Computer Science, Programming and Computational Thinking in Pre-University Studies]», Educ. Knowl. Soc., vol. 18, no. 2, pp. 7-17, 2017, [Online]. Available in: http://revistas.usal.es/index.php/eks/article/view/eks2017182717/17445%0Ahttp://rua.ua.es/dspace/handle/10045/68598.
[7]M. Zapata-Ros, «Computational thinking: A new digital literacy», Rev. Educ. a Distancia, vol. 46, n.o 46, 2015, doi: 10.6018/red/46/4.
[8]F. Alegre, J. Moreno, T. Dawson, E. E. Tanjong, y D. H. Kirshner, «Computational Thinking for STEM Teacher Leadership Training at Louisiana State University», 2020, doi: 10.1109/RESPECT49803.2020.9272455.
[9]K. Harangus y Z. Kátai, «Computational thinking in secondary and higher education», 2020, vol. 46, pp. 615-622, doi: 10.1016/j.promfg.2020.03.088.
[10]T. S. C. da Silva, J. C. B. de Melo, y P. Tedesco, «The creative process in the development of computational thinking in higher education», 2020, vol. 1, pp. 215-226, [En línea]. Disponible en: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85091441415&partnerID=40&md5=ed29efdb47cec9249a9c71fcc5e58648.
[11]J. Wu, Y. Wang, H. Kong, y L. Zhu, «How to cultivate computational thinking-enabled engineers: A case study on the robotics class of zhejiang university», 2019.
[12]R. Delgado, S. Yacchirema, F. Coral, y J. Celi, «Platform for university teaching applying Computational Thinking at the Armed Forces University ESPE », 2019, vol. 2019-June, doi: 10.23919/CISTI.2019.8760819.
[13]J. W. Wilson, M. L. Fernandez, y N. Hadaway, «Mathematical Problem Solving», The University of Georgia, 1993. http://jwilson.coe.uga.edu/emt725/PSsyn/Pssyn.html (accedido mar. 26, 2022).
[14]C. K. Shyamala, C. S. Velayutham, y L. Parameswaran, «Teaching computational thinking to entry-level undergraduate engineering students at Amrita University», 2017, pp. 1731-1734, doi: 10.1109/EDUCON.2017.7943084.
[15]A.-D. Fang, G.-L. Chen, Z.-R. Cai, L. Cui, y L. Harn, «Research on blending learning flipped class model in colleges and universities based on computational thinking - "Database principles" for example», Eurasia J. Math. Sci. Technol. Educ., vol. 13, n.o 8, pp. 5747-5755, 2017, doi: 10.12973/eurasia.2017.01024a.
[16]J. L. Weese, «Mixed methods for the assessment and incorporation of computational thinking in K-12 and higher education», 2016, pp. 279-280, doi: 10.1145/2960310.2960347.
[17]B. Kules, «Computational thinking is critical thinking: Connecting to university discourse, goals, and learning outcomes», Proc. Assoc. Inf. Sci. Technol., vol. 53, n.o 1, pp. 1-6, 2016, doi: 10.1002/pra2.2016.14505301092.
[18]S. Papert, MINDSTORMS Children, Computers, and Powerful Ideas. Estados Unidos, 1980.
[19]L. Barrera, "Algorithms and programming for teaching and learning school mathematics.", J. Chem. Inf. Model., vol. 53, no. 9, pp. 1689-1699, 2013, [Online]. Available in: http://funes.uniandes.edu.co/18563/.
[20]M. Resnick et al., «Scratch: Programming for all», Commun. ACM, vol. 52, n.o 11, pp. 60-67, nov. 2009, doi: 10.1145/1592761.1592779.
[21]S. Bocconi et al., Developing Computational Thinking : Approaches and Orientations in K-12 Education, n.o June. 2016.
[22]V. Barr y C. Stephenson, «Bringing computational thinking to K-12», ACM Inroads, vol. 2, n.o 1, p. 48, 2011, doi: 10.1145/1929887.1929905.
[23]P. J. Denning, "What is Computation?", 2010. [Online]. Available in: http://ubiquity.acm.org1.
[24]J. Voogt, P. Fisser, J. Good, P. Mishra, y A. Yadav, «Computational thinking in compulsory education: Towards an agenda for research and practice», Educ. Inf. Technol., vol. 20, n.o 4, pp. 715-728, 2015, doi: 10.1007/s10639-015-9412-6.
[25]P. J. Denning et al., «Report on the ACM Task Force on the Core of Computer Science», New York, 1988.
[26]C. Puhlmann, "DEVELOPMENT OF COMPUTATIONAL THINKING THROUGH UNPLUGTED ACTIVITIES IN BASIC EDUCATION", Federal University of Rio Grande do Sul, 2017.
[27]Megan Smith, «Computer Science For All | whitehouse.gov», 2016. https://obamawhitehouse.archives.gov/blog/2016/01/30/computer-science-all (accedido jun. 15, 2019).
[28]R. I. Dorn, K. Kanikeberg, y G. Mendoza, «Computer Science K–12 Learning Standards», EEUU, 2018.
[29]C. and D.-E. Ministry of Education, "Programming, robotics and computational thinking in the classroom", Spain, 2018.
[30]Europe 2020 Strategy, «European Commission—European Commission», 2018. https://ec.europa.eu/info/business-economy-euro/economic-and-fiscal-policy-coordination/eu-economic-governance-monitoring-prevention-correction/european-semester/framework/europe-2020-strategy_en (accedido jun. 11, 2019).
[31]A. Balanskat y K. Engelhardt, «Computing our future: Computer programming and coding. Priorities,

school curricula and initiatives across Europe», 2015. [En línea]. Disponible en: http://www.eun.org/c/document_library/get_file?uuid=3596b121-941c-4296-a760-0f4e4795d6fa&groupId=43887.
[32]S. Bocconi, A. Chioccariello, y J. Earp, «The Nordic approach to introducing computational thinking and programming in compulsory education», 2018, doi: 10.17471/54007.
[33] Brazilian Computer Society, "Guidelines for Teaching Computing in Basic Education Teaching", Brazil, 2019. [En línea]. Disponible en: http://www.sbc.org.br/documentos-da-sbc/send/131-curriculos-de-referencia/1177-diretrizes-para-ensino-de-computacao-na-educacao-basica.
[34] Innovation Center for Brazilian Education, "Reference Curriculum in Technology and Computing", Brazil, 2018. [En línea]. Disponible en: http://curriculo.cieb.net.br/.
[35]R. Paucar-Curasma y H. Paucar-Curasma, «Computational thinking in school education in South America: Systematic review of the literature», Proc. 2020 IEEE 27th Int. Conf. Electron. Electr. Eng. Comput. INTERCON 2020, pp. 9-12, 2020, doi: 10.1109/INTERCON50315.2020.9220200.
[36]R. Paucar-Curasma, N. Jara, H. Paucar-Curasma, y V. Cruz, «Assessment of Computational Thinking in regular basic education: Case I.E.T.P. "josé Obrero"», Proc. 2019 IEEE 26th Int. Conf. Electron. Electr. Eng. Comput. INTERCON 2019, pp. 1-4, 2019, doi: 10.1109/INTERCON.2019.8853613.
[37]R. Paucar-Curasma, K. O. Villalba-condori, N. Jara, R. Quispe, J. Cabrera, y M. Ponce-Aguilar, «Computational Thinking and Block-Based Programming for Beginning Engineering Students : Systematic Review of the Literature», 2021 XVI Lat. Am. Conf. Learn. Technol., pp. 530-533, 2021.
[38] R. Paucar-Curasma, K. Villalba-Condori, D. Arias-Chavez, N.-T. Le, G. Garcia-Tejada, I. Frango-Silveira, Evaluación del pensamiento computacional utilizando cuatro robots educativos con estudiantes de primaria en perú, Education in the Knowledge Society (EKS) 23 (2022). URL: https://revistas.usal.es/tres/index.php/eks/article/view/26161. doi:10.14201/eks.26161.
[39] K. O. Villalba-Condori, S. E. C. Cuba-Sayco, E. P. G. Chávez, C. Deco, C. Bender, Approaches of learning and computational thinking in students that get into the computer sciences career, in: Proceedings of the Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality, TEEM'18, Association for Computing Machinery, New York, NY, USA, 2018, p. 36–40. URL: https://doi.org/10.1145/3284179.3284185. doi:10.1145/3284179.3284185..
[40]UK Bebras Computational Thinking Challenge, «What is Bebras | www.bebras.org», 2015. https://www.bebras.org/ (accedido sep. 05, 2020).
[41]A. Rojas-López y F. J. García-Peñalvo, «Assessment of computational thinking skills to predict student learning and retention in the subject programming computer in higher education», Rev. Educ. a Distancia, vol. 20, n.o 63, 2020, doi: 10.6018/RED.409991.
[42]Talent Search, «Computer Olympiad – Computer Olympiad South Africa», 2015. http://olympiad.org.za/ (accedido sep. 05, 2020).
[43]M. Santos, «The Solving of Mathematical Problems: Advances and Perspectives in the Construction of a Research and Practice Agenda», Investig. in Educ. matemática XII, n.o 1985, pp. 1-24, 2008.
[44]B. Ortega and M. Asensio, «Evaluating Computational Thinking through Problem Solving: Validation of an Evaluation Instrument», Rev. Iberoam. Educ.-Evaluation, vol. 14, no. 1, pp. 153-171, 2021, doi: 10.15366/riee2021.14.1.009.
[45]E. Jiménez, A. Servetto, and P. Calvo, "Desarrollo de Pensamiento Computacional en Estudiantes de Ingeniería", Universi Nac. la Plata, pp. 1-10, 2020.
[46]A. Garcia-Holgado, C. Deco, N. Bredegal-Alpaca, C. Bender, K. O. Villalba-Condori, Perception of the gender gap in computer engineering studies: a comparative study in peru and argentina, in: 2020 IEEE Global Engineering Education Conference (EDUCON), 2020, pp. 1252–1258. doi:10.1109/EDUCON45650.2020.9125224.
[47]K. O. Villalba Condori, R. Avello MartÃnez, Resiliencia como factor determinante para la satisfacciÃcon la vida en estudiantes universitarios, EducaciÃMÃSuperior 33 (2019). URL: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21412019000300007&nrm=iso.
[48] L. M. Oliva Córdova, H. R. Amado-Salvatierra, K. O. Villalba Condori, An experience making use of learning analytics techniques in discussion forums to improve the interaction in learning ecosystems, in: P. Zaphiris, A. Ioannou (Eds.), Learning and Collaboration Technologies. Designing Learning Experiences, Springer International Publishing, Cham, 2019, pp. 64–76.