Eurasian Journal of Educational Research 119 (2026) 1-30

Eurasian Journal of Educational Research
www.ejer.com.tr

Pedagogical approaches for promoting computational thinking in educational settings: a semi-systematic literature review

Luis Andres Delgado[1]*, Jesus Insuasti[2]

ARTICLE INFO

Article History:
Received: 25 November 2025
Received in revised form: 20 March 2026
Accepted: 30 March 2026
DOI: 10.14689/ejer.2026.119.01

Keywords
Pedagogical Approaches, Trends, Educational Technology, Semi-Systematic Review.

ABSTRACT

Purpose. Computational Thinking (CT) is increasingly recognized as a key 21st-century skill that contributes to the development of problem-solving, creativity, and logical reasoning abilities across various disciplines. This raises the question: How is CT being promoted, and what pedagogical approaches and trends are being used in its implementation? **Methodology.** Within this framework, the present study analyzes the literature through a semi-systematic review that incorporates elements of the PRISMA model in the search, selection, and analysis of studies from different academic databases. This allows us to understand how CT is addressed in the educational field. **Results.** The findings reveal the presence of several approaches, with block-based programming and educational robotics being the most recurrent, as well as gamification, virtual environments and blended learning, and offline CT, the latter becoming a viable alternative in contexts with limited technological resources. Furthermore, a trend toward the integration of approaches is identified, suggesting that the development of the CT is moving toward a multimodal model that articulates digital innovation with pedagogical foundations. **Implications for research and practice.** Finally, the study highlights the importance of designing context-sensitive strategies that address educational equity, teacher training, and curricular coherence in the implementation of the CT at all academic levels.

Introduction

Today, we humans are immersed in technology, which is transforming various aspects of society, including education, which is facing a redefinition of teaching and learning processes. In this context, it is essential to develop new skills that allow students to meet the challenges of the 21st century. One of the most prominent is Computational Thinking (CT), which provides tools such as decomposition, abstraction, pattern identification, and algorithm design (Wing, 2006). These concepts and processes are inherent to computer science, enabling a structured and efficient approach to problems. However, despite its relevance, incorporating CT into the classroom presents several challenges, such as teacher

[1] Doctorate in Pedagogy, Mariana University, Pasto 520001, Colombia.
ORCID: https://orcid.org/0009-0005-0591-1254 , Email: luis.delgado221@umariana.edu.co
[2] University of Nariño, Doctoral Thesis Adviser, Mariana University, Pasto 520001, Colombia
ORCID: https://orcid.org/0000-0002-3233-3577 , Email: insuasti@udenar.edu.co
*Correspondence: luis.delgado221@umariana.edu.co

training, access to technological resources and the internet, resistance to pedagogical change, and the obsolescence of teaching models. This highlights the need to design strategies that promote the understanding and application of CT.

In this context, the need arises to analyze how computer programming (CP) is being promoted in the educational field and what pedagogical approaches are being used for its development. Therefore, this article presents a semi-systematic literature review to identify and analyze these approaches, which include block-based programming, educational robotics, gamification, virtual learning environments and blended learning, offline CP, and problem-solving strategies. This review aims to identify trends, contributions, and gaps in the analyzed literature, which includes research published between 2011 and 2025, with an emphasis on studies developed in Ibero-American and Latin American contexts, complemented by international contributions.

Theoretical framework of computational thinking

The integration of information and communication technologies (ICTs) has generated significant changes in various areas of human development, particularly in education, where training processes are being revolutionized. In this context, the need arises to develop new skills that enable students to face the challenges of a digital society, among which Computational Thinking has become one of the most relevant. In this sense, this theoretical framework addresses Computational Thinking from its conceptual foundations, main frames of reference, integration into education, and its incorporation into the curriculum.

Conceptual basis

Computer-based learning (CBL) has been approached from different perspectives that allow us to understand its scope in the teaching and learning process. The contributions of Seymour Papert (1980), from a constructivist perspective, highlight the role of active learning mediated by technology through procedural thinking and computer literacy, where students construct their knowledge through interaction with computational environments. Although the author does not explicitly define CBL, his postulates lay the groundwork for understanding how technological tools can contribute to changing the patterns of access to and construction of knowledge.

From this perspective, formally introduces the concept of CT, defining it as a way of thinking oriented towards problem-solving, systems design, and process understanding based on computer Wing (2006) science concepts. The author argues that this skill is not limited to the field of programming, but can be applied in different contexts and disciplines, considering it a fundamental competence for everyone that facilitates addressing situations in a logical and organized manner.

Subsequent studies have broadened the concept, and it is now recognized as a set of cognitive skills. Authors such as Grover and Pea (2013) suggest that cognitive programming involves processes of abstraction, decomposition, and algorithmic thinking, which allow for problem-solving in different contexts.

CT Reference Frames

The CT has not only been addressed from its conceptual definition, but also from different reference frameworks that provide structured guides that allow organizing its components, processes and forms of application in the classroom. Along these lines, Wing (2010) expands his work by highlighting its transversal nature, considering it a 21st-century literacy that allows one to understand which elements of a problem can be addressed through computing, evaluate the relationship between available techniques and tools, and apply computational strategies to address problems in different contexts.

In line with this approach, in 2011, the International Society for Technology in Education — ISTE — which is an organization dedicated to promoting the use of technology in education, and the Computer Science Teachers Association — CSTA —, proposed a framework in which they define CT as a problem-solving approach in which situations are posed that can be addressed using computers or other technological artifacts, where data are organized, analyzed and represented through abstractions such as models and simulations, automating solutions through algorithmic thinking and generalizing these processes in different contexts (Sociedad Internacional para la Tecnología en la & Asociación de Profesores de, 2011).

In this regard, Brennan and Resnick (2012) developed the *computational thinking framework*, which comprises three dimensions: computational concepts, involving programming elements such as iteration and parallelism; computational practices, related to the processes involved in programming; and computational representations, which are developed about their environment and themselves because of their programming. This model allows for a more comprehensive understanding of computational thinking by articulating content, processes, and forms of interaction in learning.

Complementarily, other studies have contributed to the conceptualization of computational thinking from different perspectives. Roig-Vila and Moreno-Isac (2020), based on a literature review, demonstrate that computational thinking can be approached from various angles, highlighting that assessment is a key element in its development and that it is necessary to design and validate instruments that allow for its measurement in educational contexts. Along the same lines, Polanco Padrón et al. (2020) propose that computational thinking is primarily linked to mental processes such as logic and abstraction, as well as to the ability to identify which elements of a problem can be addressed through computation, understand its limitations and potential, apply or innovate tools or techniques for new requirements, recognize opportunities to use computation in innovative ways, and employ specific computational strategies in the current context.

Likewise, Moreno et al. (2019) define computer science as the ability to formulate and model problems using computer science concepts and tools, expanding its understanding as a structured process oriented toward problem-solving. Based on these contributions, it is evident that computer science is not limited to a single definition, but rather its conceptualization links skills, processes, and frameworks that guide its development and incorporation into educational processes.

CT in education

In the educational context, computer literacy has gained increasing relevance as it has been incorporated into teaching and learning processes, considered a competency that promotes the development of cognitive skills in diverse areas of knowledge. Its integration is not limited to teaching programming itself, but rather relates to how students understand, represent, and approach diverse situations, fostering logical reasoning, scenario modeling, and the design of creative strategies, thus helping to prepare students to participate in and face the challenges of the digital society, promoting adaptability and technological fluency (Grover & Pea, 2013).

In this sense, the scope of CT is not limited to programming and coding; it goes beyond that, becoming a set of cognitive skills that students can apply in different contexts. As Korkmaz and Yıldız (2020) point out, this type of thinking promotes analytical reasoning, decision-making, and the development of higher-order thinking skills such as analyzing, evaluating, and creating, allowing students to approach problems systematically and design efficient solutions in both academic settings and everyday life.

From this perspective, computer science pedagogy focuses on the use of methodologies and approaches that facilitate learning processes in the classroom and contribute to the achievement of educational objectives. In this sense, its integration into curricula is consolidated as a strategy for the holistic development of students. In this regard, Ortega Ruipérez (2018) proposes pedagogical guidelines that include programming, technology-free activities, problem-solving, and the development of interdisciplinary projects, demonstrating the diversity of strategies that can be employed in its teaching.

The promotion of computer literacy in the classroom is linked to the need to respond to the demands of an increasingly media-driven society. Zapata-Ros (2015) argues that the development of these skills allows students to access information and use it efficiently to solve academic and social problems through divergent and reflective thinking, while Balladares Burgos et al. (2016) emphasize that it is not enough to learn how to use technological tools; it is necessary to integrate them into the processes of knowledge creation, innovation, and management.

In this context, societal transformations require innovative pedagogical approaches, where teaching strategies are grounded in theories that delve into the teaching and learning processes and adapt to the specific realities of students. This implies promoting actions that enable students to understand, intervene in, and resolve situations using available resources (Mallart et al., 2001). Therefore, it is necessary to strengthen computer skills and mastery of the cognitive processes associated with this type of thinking. Otherwise, as Ortega-Ruipérez and Asensio Brouard (2018) point out, their application could be limited to solving computer-related problems.

Based on the above, the need to analyze different strategies, approaches and resources that have been used for the development of CT in the classroom is evident, to understand not only its scope, but also its particularities that characterize its implementation in the training processes.

Incorporation of the CT into the curriculum

One of the challenges in education systems is the integration of computer literacy, which goes beyond its conceptual definition or methodological approach, as it implies changes in the curriculum, pedagogical practice, and educational policies. In this sense, its integration requires not only the inclusion of thematic content but also a rethinking of teaching and learning processes oriented toward the development of cognitive skills and problem-solving in different contexts.

From this perspective, the incorporation of Computer Science into the curriculum has been approached from different angles. Sánchez Vera (2019) identifies two main approaches: first, as a specific subject with an emphasis on technology, and second, as a cross-curricular theme integrated with other areas of knowledge. Furthermore, Barr and Stephenson (2011) defines guidelines that direct the integration of this approach into educational systems, as well as its evaluation, teacher training, and access to the necessary resources. This allows us to understand that implementing Computer Science is not only a change in the curriculum but also an institutional commitment that encompasses both pedagogical and administrative aspects, thus constituting a comprehensive process.

Along these same lines, studies such as García (2020) show that the incorporation of computer-based learning (CBL) in the classroom is voluntary and depends on factors such as institutional participation, resource availability, and teacher motivation. Sotelo Martín (2022), for his part, highlights teachers' willingness to integrate CBL into the classroom, as well as the need for training processes for its proper implementation. Thus, the incorporation of the CT is configured as a process that involves the articulation of curricular, pedagogical, and institutional elements. This highlights the need to move towards comprehensive and contextualized proposals for its implementation, considering the particularities and conditions in which teaching and learning processes take place.

The article is organized into six sections: the first is the introduction and the theoretical framework also highlighting the few empirical studies as well as an overview of the definition of CT and its application in educational settings; the second describes the methodology; the third presents the results; the fourth develops the discussion; and the final section presents the general conclusions of this study.

Methodology

Research design

This article was developed from a semi-systematic literature review aimed at identifying and analyzing the pedagogical approaches and perspectives used in promoting personal development in the educational field. This approach was chosen due to the diversity of methodologies, contexts, teaching methods, and strategies present in the analyzed studies, which makes it difficult to apply strictly comparative criteria typical of a systematic review. Therefore, although the research is not framed as a systematic review in the strict sense, it incorporates some methodological elements of the PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) approach, particularly in the identification, screening, eligibility, and inclusion phases, with the aim of providing greater coherence and rigor to the study.

Data collection

The search for information was carried out in the period between the years 2011 and 2025, for these academic databases and digital repositories were used, such as Scopus, ERIC, Scielo, Redalyc and Google Scholar, in which emphasis was placed on scientific articles, doctoral theses and academic works in Ibero-America and Latin America, including some international publications.

The primary search criterion used was the term "computational thinking," which was combined with "problem-solving," "pedagogical approaches," and "educational context." Boolean operators such as AND and OR were used to combine terms or include alternatives in the results. Based on this, 148 documents were identified, of which 79 were selected after reading their abstracts, and 53 were then chosen for the final analysis. This selection process is presented in Figure 1 using a PRISMA-based flowchart.

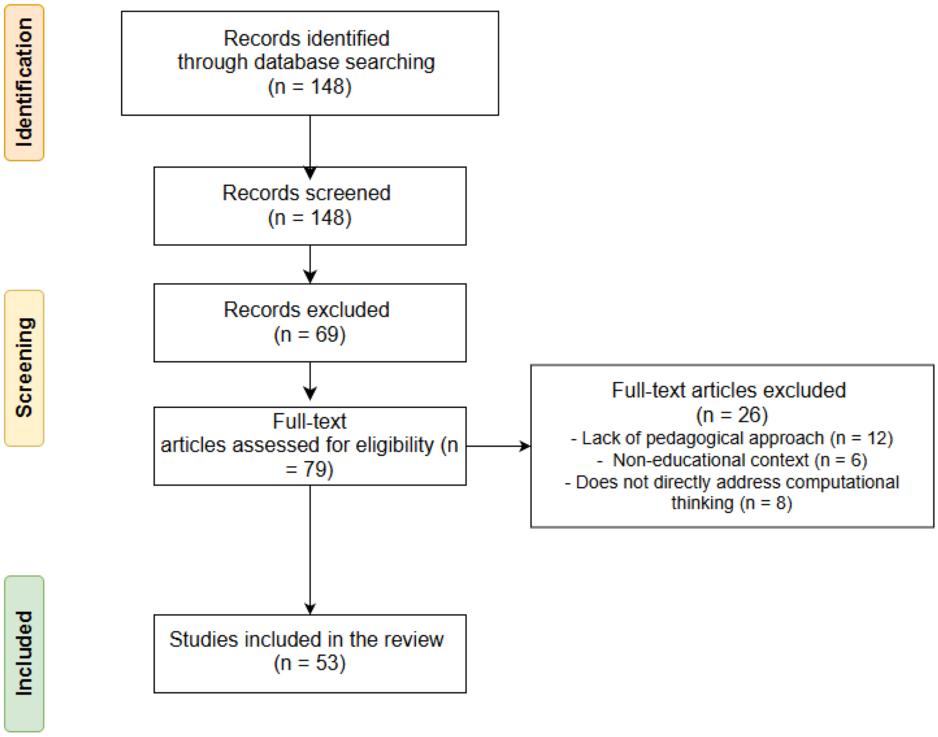

Figure 1: *Flowchart of the study selection process based on PRISMA*

Inclusion and exclusion criteria

Inclusion criteria were applied, including research focused on teaching or promoting problem-solving in education, studies aimed at developing problem-solving skills, and works that describe or implement experiences in the use of teaching resources related to pedagogical approaches in this field. Articles without a pedagogical focus, those developed in contexts other than education, or those that did not directly address problem-solving were excluded.

Data analysis

To organize and analyze the 53 selected studies, a categorization process was defined where each study constituted the unit of analysis. In this sense, the studies were organized into two dimensions: pedagogical approaches and theoretical foundations. The first grouped research focused on the implementation of the CT in the classroom, while the second encompassed studies oriented towards its conceptualization, components, and frameworks.

As a result of this process, 38 studies were classified within the pedagogical approaches dimension. These studies were analyzed based on the strategies and resources used, identifying six categories: block programming, educational robotics, gamification, virtual learning environments and blended learning, offline PCs, and problem-solving approaches. In studies that addressed several categories, the most prevalent category, that is, the one with the greatest emphasis, was selected. The remaining 15 studies were classified within the theoretical foundations dimension and were used to strengthen the conceptual framework and inform the discussion. Therefore, although the final corpus of the review consisted of 53 studies, Table 2 and Figure 2 present only 38 studies classified within the pedagogical approaches dimension.

Table 1 presents 53 studies included in this study, which comprises information related to the author(s), year of publication, title, educational level, type of study, main result, and contribution to the study, forming the corpus of this review. The analysis was carried out through a qualitative synthesis in which the studies were organized and examined according to the defined dimensions and categories. Based on this, the main characteristics, teaching strategies, and contributions of each approach to promoting CT in the educational field were described, allowing for the identification of general trends in its implementation.

Table 1

Studies included in the review

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
1.	Alsina & Acosta	2022	Connecting early childhood mathematics education and computational thinking: learning repeating patterns with the Cubetto programmable educational robot	Early Childhood Education	Empirical	Development of mathematical skills and computational thinking	Pedagogical Approach
2.	Álvarez-Herrero, JF	2020	Computational thinking in Early Childhood Education, beyond floor robots	Early Childhood Education	Empirical	Educational robotics promotes CT development but needs to be complemented with other pedagogical approaches.	Pedagogical Approach
3.	Auchstetter et al.	2023	Evaluation of The Pack: Integrating a Computational Thinking Game in Middle School Classrooms	Secondary Education	Empirical	Gamification through digital games improves the resolution of computational problems.	Pedagogical Approach
4.	Balladares Burgos et al.	2016	From complex thinking to computational thinking: challenges for contemporary education	General	Theorist	The CT is presented as an evolution of complex thinking in education	Theoretical Foundation
5.	Barr & Stephenson	2011	Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?	K-12	Theorist	CT integration requires systemic changes	Theoretical Foundation
6.	Basogain-Olabe et al.	2015	Computational Thinking through Programming: Learning Paradigm	General	Theorist	Programming is a fundamental means for CT development	Pedagogical Approach

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
7.	Bell et al.	2011	Computer science unplugged: school students doing real computing without computers	Secondary Education	Empirical	The CT can be developed without technology through structured activities	Pedagogical Approach
8.	Brennan & Resnick	2012	New frameworks for studying and assessing the development of computational thinking	General	Theorist	CT dimensions are defined: concepts, practices and perspectives	Theoretical Foundation
9.	Bruges & Camperos	2021	Influence of computational thinking on the cognitive skills inherent in problem-solving methods	Higher education	Empirical	The CT improves cognitive skills in problem-solving	Pedagogical Approach
10.	Cabra Páez & Ramírez	2021	Development of computational thinking and mathematical skills in problem analysis and solving: a learning experience with Scratch on the Moodle platform	Secondary Education	Empirical	Scratch strengthens the CT and improves problem-solving	Pedagogical Approach
11.	Chiazzese et al.	2018	Engaging primary school children in Computational Thinking: designing and developing video games	Primary Education	Empirical	Video game development improves CT performance and motivation	Pedagogical Approach
12.	Eguíluz Morán, A.	2020	Analysis of the development of Computational Thinking with generalization through visual programming challenges	General	Empirical	Visual programming through challenges improves generalization in computational thinking	Pedagogical Approach
13.	Gamito et al.	2022	The development of computational thinking in education: an assessment based on an experience with Scratch	Teacher training	Empirical	Scratch promotes the development of algorithmic thinking and its application in the classroom	Pedagogical Approach

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
14.	García Rodríguez, A.	2022	Teaching programming through Scratch for the development of computational thinking in basic secondary education	Secondary Education	Empirical	Teaching with Scratch significantly strengthens computational thinking in students	Pedagogical Approach
15.	García, JM	2020	The expansion of Computational Thinking in Uruguay	General	Empirical	The expansion of computer programming in the education system is evident, with programming predominating.	Theoretical Foundation
16.	García-Valcárcel & Caballero	2019	Robotics to develop computational thinking in Early Childhood Education	Early Childhood Education	Empirical	Educational robotics significantly improves computational thinking in children	Pedagogical Approach
17.	Gómez Andrade et al.	2022	Project to strengthen computational thinking through a strategy to improve programming skills in ICT engineering students	Higher education	Empirical	Using Scratch strengthens programming skills and computational thinking.	Pedagogical Approach
18.	González-González, CS	2020	Computational thinking and robotics in early childhood education: an inclusive methodological proposal	Early Childhood Education	Empirical	Educational robotics promotes the development of computational thinking in inclusive contexts	Pedagogical Approach
19.	Grover & Pea	2013	Computational Thinking in K–12: A Review of the State of the Field	General	Theorist	Computational thinking is becoming established as a key competency in K-12 education	Theoretical Foundation

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
20.	Gutiérrez Rodríguez, CA	2018	ICT strategy to strengthen problem-solving skills through computational thinking	Secondary Education	Empirical	The use of ICT and PBL improves problem-solving and student motivation	Pedagogical Approach
21.	Huerta Jiménez & Velázquez Albo	2021	Computational thinking as a generic skill: a systematic review	General	Theorist	The CT is recognized as a transversal skill applicable in multiple areas of knowledge	Pedagogical Approach
22.	ISTE & CSTA	2011	Operational Definition of Computational Thinking for K–12 Education	General	Theorist	Standards and key definitions are established for the integration of CT in K-12 education	Theoretical Foundation
23.	Korkmaz & Yıldız	2020	The effect of computational thinking on students' problem-solving skills and academic performance: a meta-analysis	General	Theorist	Computational thinking improves cognitive skills and academic performance	Theoretical Foundation
24.	Laura-Ochoa & Bedregal-Alpaca	2021	Analysis of programming environments for the development of computational thinking skills and teaching programming to beginners	Higher education	Empirical	Block-based programming environments facilitate initial learning and CT development	Pedagogical Approach
25.	Mallart et al.	2001	Didactics: concept, object and purposes.	General	Theorist	The importance of didactics in the teaching-learning processes is highlighted.	Theoretical Foundation

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
26.	Mancinas González & Mendoza Montijo	2021	Computational thinking and adaptive learning in solving problems with fractions	Secondary Education	Empirical	Adaptive learning promotes problem-solving and CT development	Pedagogical Approach
27.	Mantilla Güiza	2021	Proposal for the development of computational thinking within a digital ecosystem. Case study: Vicente Azuero Technical School in Colombia	Secondary Education	Empirical	The use of digital ecosystems significantly improves CT dimensions	Pedagogical Approach
28.	Molina Ayuso et al.	2020	Problem-solving based on Polya's method using computational thinking and Scratch with secondary school students	Secondary Education	Empirical	Improve problem-solving using Scratch and Polya's method	Pedagogical Approach
29.	Montes-León et al.	2020	Improving Computational Thinking in High School Students with Unplugged Homework	Secondary Education	Empirical	Unplugged activities improve logical reasoning and programming learning	Pedagogical Approach
30.	Moreno et al.	2019	It's not the same: a text network analysis of computational thinking definitions to study their relationship to computer programming	General	Theorist	Computational thinking and programming are not equivalent, although they are related.	Theoretical Foundation
31.	Mujica Statzewitch	2021	Evaluation of the development of computational thinking in primary and secondary education students	Primary/Secondary Education	Empirical	A moderate level of CT development is evident in students, with progressive improvement by level	Pedagogical Approach
32.	Ortega & Asensio	2021	Assessing Computational Thinking through Problem Solving: Validation of an Assessment Instrument	Secondary Education	Empirical	A reliable instrument for evaluating computational thinking is validated	Pedagogical Approach

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
33.	Ortega Ruipérez	2018	Computational thinking and problem solving	Secondary Education	Empirical	The CT significantly improves the resolution of complex problems.	Pedagogical Approach
34.	Ortega-Ruipérez & Asensio	2018	DIY Robotics: Computational Thinking to Improve Problem Solving	Secondary Education	Empirical	The CT is more easily applied in the execution of the algorithm than in the representation of the problem.	Theoretical Foundation
35.	Papert, S.	1980	Mindstorms: Children, computers, and powerful ideas	General	Theorist	Programming promotes meaningful learning and the development of logical thinking	Theoretical Foundation
36.	Polanco Padrón et al.	2020	An approach to a definition of computational thinking	General	Theorist	The CT is conceived as a mental process based on logic and problem-solving	Theoretical Foundation
37.	Rico & Basogain	2018	Computational thinking: breaking down digital and educational divides	General	Empirical	Integrating the CT into the curriculum helps to reduce digital and educational divides.	Pedagogical Approach
38.	Roig-Vila & Moreno-Isac	2020	Computational thinking in education. Bibliometric and thematic analysis.	General	Theorist	There is a growing body of scientific research on CT with diverse approaches and methodologies	Theoretical Foundation

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
39.	Rojas López	2019	Personalized learning scenarios based on the assessment of computational thinking for learning programming skills through a b-Learning and gamification environment	Higher education	Empirical	Personalized learning and gamification reduce dropout rates and improve motivation	Pedagogical Approach
40.	Roncoroni & Bailón	2020	Computational thinking. Digital literacy without computers	General	Theorist	The CT can be developed without computers, integrating cultural and educational dimensions.	Pedagogical Approach
41.	Sánchez Vera	2019	Computational Thinking in Educational Environments: An Approach from Educational Technology	General	Theorist	The CT requires a cross-cutting pedagogical approach for its integration into educational contexts	Theoretical Foundation
42.	Sandoval et al.	2020	SCRATCH as a tool for developing computational thinking in basic education	Basic Education	Empirical	Scratch significantly improves logical thinking and problem-solving skills.	Pedagogical Approach
43.	Sotelo Martín	2022	Computational Literacy and Thinking in Teachers: A Descriptive Study	Basic education (teachers)	Empirical	There is evidence of low teacher training in programming and computational thinking	Theoretical Foundation
44.	Terroba et al.	2021	Cultivating mathematical talent in Early Childhood Education through problem-solving to promote the development of computational thinking	Early Childhood Education	Empirical	Improve motivation and problem-solving skills through robotics	Pedagogical Approach

S.No	Author(s)	Year	Qualification	Level	Type of study	Main result	Contribution to the study
45.	Terroba et al.	2021	Computational thinking in solving contextualized problems in a story in Early Childhood Education	Early Childhood Education	Empirical	Improves cognitive skills and learning in a narrative context	Pedagogical Approach
46.	Triantafyllou et al.	2024	Students' development of Computational Thinking and Teachers' professional development via Bebras through Gamification: A systematic literature review	Secondary	Theorist	Gamification promotes CT development	Pedagogical Approach
47.	Urquizo	2020	Incorporating Computational Thinking in Engineering as support for the Problem-Solving competency: playing with Lightbot	Higher education	Empirical	Improves problem-solving skills	Pedagogical Approach
48.	Vázquez Uscanga et al.	2019	Computational thinking in the classroom: the challenge in Latin American education systems	General	Theorist	The implementation of the CT depends on educational policies	Pedagogical Approach
49.	Velásquez	2022	Development of computational thinking in early childhood	Early Childhood Education	Empirical	Predominance of algorithmic thinking in children	Pedagogical Approach
50.	Yang	2025	A three-phase professional development approach to improving robotics pedagogical knowledge and computational thinking attitude of early childhood teachers	Teacher training	Empirical	Improves pedagogical knowledge and attitudes towards the CT	Pedagogical Approach
51.	Yılmaz and İzmirli	2023	Effect of unplugged and plugged coding activities on secondary school students' computational thinking skills	Secondary	Empirical	Both approaches improve CT without significant differences	Pedagogical Approach
52.	Zapata-Ros	2019	Unplugged computational thinking	Early Childhood / Primary	Theorist	The CT can be developed without technology	Pedagogical Approach
53.	Zhang et al.	2025	Developing preschool children's computational thinking and executive functions: unplugged vs. robot programming activities	Early Childhood Education	Empirical	Robotics surpasses the unplugged approach to PCs and executive functions	Pedagogical Approach

Results

The literature review focused on identifying the pedagogical approaches used to promote and develop computer programming (CP) in education. From the 53 studies classified under this dimension, and because of the categorization process, six main approaches were identified: block programming, educational robotics, virtual learning environments and blended learning, gamification, offline CP, and problem-solving approaches. These approaches structure the organization of the results in this review. The distribution of studies according to these categories is presented in Table 2.

Table 2

Distribution of studies classified within the pedagogical approaches dimension (2011-2025, n=38)

Approach	Number of studies	Period	Authors
Block programming	10	2018-2022	Basogain-Olabe et al. (2015); Cabra Páez and Ramírez Gamboa (2021); Gamito et al. (2022); García Rodríguez (2022); Gómez Andrade et al. (2022); Laura-Ochoa and Bedregal-Alpaca (2021); Mancinas González and Montijo Mendoza (2021); Molina Ayuso et al. (2020); Rico and Basogain Olabe (2018); Sandoval et al. (2020)
Educational Robotics	9	2019-2025	Alsina and Acosta (2022); Álvarez Herrero (2020); García-Valcárcel-Muñoz-Repiso and Caballero-González (2019); González González (2020); Mujica de Statzewitch (2021); Terroba Acha, Ribera, et al. (2021); Terroba Acha, Ribera Puchades, et al. (2021); Yang (2025); Zhang et al. (2025)
Virtual environments and blended learning	3	2018-2021	Gutiérrez Rodríguez (2018); Mantilla Güiza (2021); Rojas López (2019)
Gamification	5	2018-2024	Auchstetter et al. (2023); Chiazzese et al. (2018); Eguíluz (2020); Triantafyllou et al. (2024); Urquizo et al. (2021)
CT disconnected	6	2011-2022	Bell et al. (2009); Montes-León et al. (2020); Roncoroni and Bailón Maxi (2020); Velásquez Montero (2022); Yilmaz and İzmirli (2023); Zapata-Ros (2019)
Problem-based approaches	5	2018-2021	Brugés Romero and Camperos Villamizar (2021); Huerta Jiménez and Albo (2021); Ortega Ruipérez (2018); Ortega Ruipérez and Asensio Brouard (2021); Vázquez Uscanga et al. (2019)

The distribution of studies reveals a tendency towards recurring strategies based on the use of technological tools, particularly block-based programming and educational robotics. Furthermore, proposals such as gamification, virtual environments and blended learning, offline PCs, and problem-solving approaches that expand the possibilities for CT development were identified. Taken together, these approaches reflect the diversity of strategies employed in promoting CT and guide the analysis presented below. The organization of the studies by pedagogical approach is presented graphically in Figure 2.

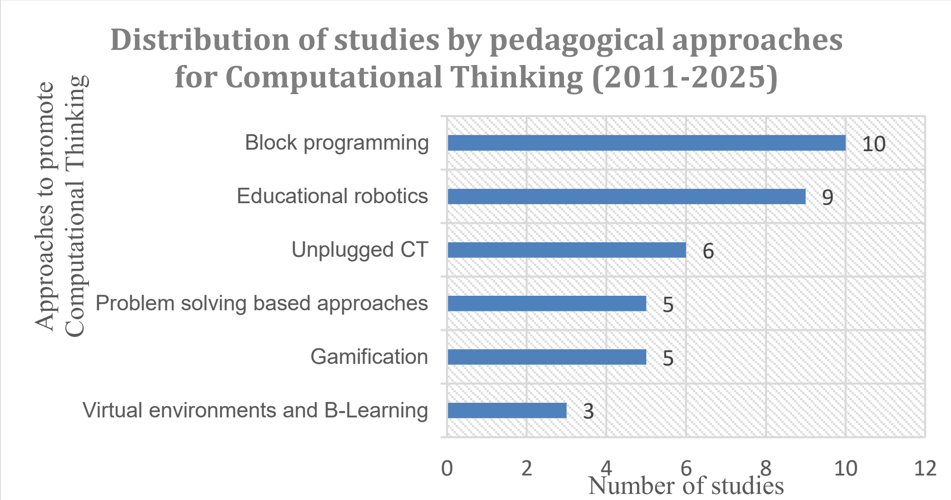

Figure 2: *Distribution of studies by pedagogical approaches for promoting Computational Thinking (2011-2025, n=38)*

Block programming

Block-based programming is one of the most widely used approaches for integrating computer skills into learning. It is a visual programming environment that uses blocks to represent instructions, helping students become familiar with fundamental computer science and programming concepts. It is also used to foster creativity, logical thinking, and problem-solving abilities.

Within this framework, a trend is observed in the use of tools like Scratch for developing computer-related skills. Along these lines, Sandoval et al. (2020) studied gamification using Scratch in basic education, adapting the LEGO methodology, which comprises the phases of connect, build, contemplate, and continue. This allowed for the structuring of activities aimed at improving analysis, conceptualization, and problem-solving skills. This approach aligns with the work of Molina Ayuso et al. (2020), who applied Polya's method, which consists of four steps: understand the problem, devise a plan, carry it out, and check the solution. Their study, using Scratch, demonstrated that approximately 70% of the participants improved in these skills, thus confirming the link between block-based programming and mathematical thinking. Along these same lines, Cabra Páez and Ramírez Gamboa (2021) designed a Scratch-supported course within the Moodle platform, aimed at promoting mathematical analysis and problem-solving within the STEM (Science, Technology, Engineering, and Mathematics) framework, demonstrating a significant impact on the development of these skills. These studies agree on using block-based programming as a strategy to strengthen problem-solving, linking computer science with mathematical reasoning through structured methodologies and tools like Scratch.

Regarding the incorporation of block-based programming in virtual environments, García Rodríguez (2022) proposes a pedagogical strategy based on Scratch, implemented on the Moodle platform. This strategy was used for remote learning during the COVID-19 pandemic, demonstrating improvements in computer skills among secondary school students. Similarly, Gómez Andrade et al. (2022) implemented a self-directed course on essential programming concepts, with activities developed using Scratch and hosted on the Classroom learning management platform, strengthening computer literacy and consolidating programming skills.

Furthermore, the use of block programming has also been addressed through its integration into the curriculum. Rico and Basogain Olabe (2018) describe the implementation of the course "CT-01: Introduction to Computational Thinking," an initiative of the Colombian Ministry of Information and Communication Technologies (MINTIC), in collaboration with the National Research and Education Network (RENATA), which aims to familiarize students and teachers with computational thinking concepts as one of the key competencies of the 21st century. Continuing this work, Basogain-Olabe et al. (2015) develop two pedagogical experiences: the first in Uruguay, using virtual environments and technological devices such as the BBC Micro:bit board, and the second in the Dominican Republic, where the course implemented in Colombia is being used, demonstrating that these new pedagogies have a positive impact in the classroom.

Regarding training and skills development, Gamito et al. (2022) compiled the perceptions of pre-service teachers on the development of computer programming (CP) and the educational possibilities of programming and the Scratch environment. Through practical activities and reflective processes, they demonstrated that this tool fosters the development of attitudes and skills in organization, data analysis, and the automation of solutions through algorithmic thinking. Meanwhile, Mancinas González and Montijo Mendoza (2021) implemented a mobile application to identify the types of calculations needed to solve fraction problems and the possibilities of adaptive learning. This learning was based on providing participants with different resources according to their number of attempts and sending alerts to offer timely support. These studies show how block-based programming, in addition to contributing to the development of CP skills, allows for the integration of strategies such as reflection and adaptive learning into teacher training.

In this same context, Laura-Ochoa and Bedregal-Alpaca (2021) analyze different environments such as Code.org, Lightbot, Scratch, Turtle with Python and the PSeInt pseudocode, highlighting Scratch for its contribution to the understanding of programming concepts and the transition to languages with syntax.

Taken together, these studies reveal a trend in the use of Scratch to develop computer skills, basic programming concepts, and problem-solving processes, highlighting its implementation at various educational levels and contexts. However, other tools, such as Code.org, PSeInt, Lightbot, Turtle, LEGO, Mindstorms, and App Inventor, have been explored, offering similar functionalities. Some of these focus on programming, robotics, and algorithmic thinking, demonstrating that the development of computer literacy does not depend on a single tool, but rather on how these tools are integrated with classroom activities and processes.

Educational Robotics

Educational robotics stands out as a relevant approach to promoting Computational Thinking (CT), as it incorporates programming concepts and algorithm design in a tangible and practical way.

In the context of early childhood education, several studies have addressed the use of programmable robots to develop initial programming skills. García-Valcárcel-Muñoz-Repiso and Caballero-González (2019) analyze the use of Bee-Bot as an educational resource. This bee-shaped robot allows children to perform movements and turns on mats that present different challenges and incorporate stories, fostering the creation and structuring of sequences. Similarly, González González (2020), in her doctoral thesis, presents a proposal based on the pedagogical principles of the Maker movement, learning through play, and inclusive education. This proposal involves developing activities with the Kibo robot, which is programmed with wooden blocks, to introduce children to basic programming concepts. However, Álvarez Herrero (2020) points out that, without the necessary attention, educational robotics can become a distraction or a recreational activity, losing its potential in educational processes.

Regarding the development of problem-solving and logical thinking skills, Terroba Acha, Ribera, et al. (2021); Terroba Acha, Ribera Puchades, et al. (2021) worked with the Next 1.0 robot on a sequence of mathematical problems related to spatial orientation, using instruction cards on boards and mats based on the story "Chivos Chivones" by Olalla González, following the steps proposed in Polya's method. They highlighted that creating activities with robots has great potential to promote computational thinking and problem-solving. Reinforcing this, Alsina and Acosta (2022) used the Cubetto robot to establish the relationship between mathematics education and computational thinking, confirming that this type of resource not only supports the teaching of mathematics but also contributes to the development of skills associated with computational thinking.

Along these same lines, Zhang et al. (2025) they analyze the development of cognitive function and executive functions in preschool children, comparing the use of the Matatalab robot, unplugged activities, and traditional methodologies. They found that the group using robotics showed improvement in inhibition, working memory, and cognitive flexibility compared to the other groups. These studies demonstrate how robotics is linked to mathematical reasoning and problem-solving through practical and contextualized activities and contributes to the development of more complex cognitive skills in students.

At the primary and secondary education levels, Mujica de Statzewitch (2021) analyzes the level of computer competence achieved after two years of working with robotics and programming, evaluating aspects such as spatial orientation, reasoning and problem solving, showing that the participants reach an average level with a tendency to improve.

On the other hand, some studies address educational robotics from the perspective of teacher training and its impact on teaching processes. In this context, Yang (2025) evaluates the impact of the Computational Thinking for Early Childhood Educators (CT4ECE) program, which involves three phases: learning, practicing, reflecting, and improving, using the Matatalab robot. The study demonstrates a significant impact, as it contributed to teachers' knowledge of robotics and their willingness to engage with programming.

The findings of the studies highlight how educational robotics is becoming established as an approach to promoting computer literacy through practical, situated activities, linking programming with cognitive, mathematical, and problem-solving skills. Furthermore, its impact is reflected not only in student learning but also in teacher training, enabling the incorporation of new pedagogical strategies in the classroom.

Virtual environments and blended learning

Virtual learning environments have been used as resources to promote the development of various skills and abilities in students, as they offer communication, interaction, collaboration and transmission tools that facilitate the teaching and learning processes. In this context, Gutiérrez Rodríguez (2018) implemented a course structured as an instructional script through a virtual platform, using a problem-based learning approach at the Santo Tomás Educational Institution, located in Cali, Colombia. The resource is presented as a video game with animations, interactive activities, video quizzes, and challenges, which significantly improved the performance of most of the participating students.

Following the same line of work, Mantilla Güiza (2021) developed a virtual learning ecosystem (VLE) as a teaching strategy in technology and computer science to promote computer literacy in secondary education. Based on the application of a pre-test and post-test, improvements were observed in the conceptual, practical, and perspective dimensions proposed by the MIT model from Harvard University, aimed at developing the competencies of knowing, doing, and being, respectively, confirming that virtual environments not only facilitate access to content but also influence the development of computer literacy and problem-solving skills.

In higher education, Rojas López (2019) examines the use of gamification and learning scenarios based on a diagnostic assessment of CT skills and the application of a strategy, where learning paths are defined that foster student motivation and autonomy, showing a decrease in the dropout rate compared to previous cohorts.

These studies show how virtual environments and online learning (B-Learning) are not limited to access to information but allow the design of more dynamic and adaptive learning experiences, contributing to the development of CT, problem-solving and key 21st-century skills such as autonomy, logical thinking and creativity.

Gamification

Gamification has become a relevant resource for promoting CT, as it integrates game elements, activities and dynamics into educational processes, favoring student motivation and participation.

At the primary education level, gamification has been used as a strategy to introduce programming and computer concepts through playful environments. In this regard, Chiazzese et al. (2018) proposed a pedagogical model applied to 81 students in Italy, through the construction of 3D video games developed in four phases: narrative, abstraction, automation, and analysis, using the Kodu Game Lab application. This approach demonstrated the development of computer-related skills, as well as an increased interest in programming, highlighting even greater participation from female students, confirming that this discipline is not exclusive to men.

Following this approach, Eguíluz (2020) proposes a tool called Kodetu, a challenge-based platform for teaching basic algorithm concepts. It provides an autonomous learning experience through a game that involves block programming and stores each interaction made by students. These interactions are analyzed through indicators and reports from the KodetuWin application, allowing for monitoring and evaluation of the development of the generalization competency, a component of Computer Programming (CP). Eguíluz also highlights the importance of an organized and structured process, where the student's interaction provides valuable information about their learning process. Along the same lines, Auchstetter et al. (2023) the Pack project is implemented in primary education, using a digital game and twelve lessons designed to solve computational problems in a virtual world. The objective is to restore the balance of an ecosystem, demonstrating its potential as a teaching resource that can be integrated at different points in the CP teaching process.

In higher education, Urquizo et al. (2021) use the educational game Lightbot, a programmable robot, as a tool to incorporate computer skills and strengthen algorithmic thinking, decomposition, generalization, and automation in Systems Engineering students at the National University of San Agustín, Peru, showing significant improvements in these competencies, which confirms that gamification is not limited to initial education levels, but can contribute to the development of more complex skills.

On the other hand, some studies address gamification from a broader, more cross-curricular perspective. Initiatives like Bebras have promoted the development of computer literacy through a gamified approach that allows students to work on abstraction and decomposition, as well as computer skills. Based on these educational experiences Triantafyllou et al. (2024), they highlight the role of teachers in incorporating strategies into the curriculum and the need to consider computer literacy as a cross-curricular competency.

Taken together, these studies show how gamification not only increases student motivation but also allows for the design of structured learning experiences including video games, interactive platforms, and pedagogical strategies at different educational levels that promote CT development.

CT disconnected

The disconnected approach uses various resources that do not require computers, tablets, cell phones or other electronic devices, employing recreational activities, physical materials, group dynamics, puzzles, board games and other resources to encourage the development of computer and problem-solving skills.

One of the leading initiatives that has gained international recognition is *Computer Science Unplugged,* developed by the University of Canterbury, which offers various resources for teaching computer science concepts through technology-free activities. In this regard, Bell et al. (2009) emphasize that computer science contributes to solving real-world problems, not necessarily using computers, thus broadening the understanding of computer science beyond digital environments.

In early childhood education, Velásquez Montero (2022) analyzes how children in the early education programs at the *Aprendamos del Gran Maestro Educational Center and the El Mundo de Guerra Kindergarten* develop programming skills through connected and disconnected activities, demonstrating the use of strategies such as sequencing, decomposition, and trial and error. Along the same lines, Yilmaz and İzmirli (2023)they incorporate in-person activities with the game Tospaa for primary school students, highlighting that learning programming logic through disconnected activities facilitates the transition to digital environments like Scratch.

At the secondary education level, Montes-León et al. (2020) analyze the impact of developing disconnected activities on the fundamentals of programming in students of the Hermano Miguel School in Latacunga, Ecuador, through the use of resources such as activities from the Brebas initiative, mathematical exercises and mental games, highlighting that it is possible to promote CT without electronic equipment and that disconnected activities contribute significantly to both its development and the understanding of the fundamentals of programming.

On the other hand, some studies address the unplugged approach from its incorporation into the curriculum and its relationship with other areas of knowledge. Zapata-Ros (2019) compiled three international experiences. In New Zealand, CS Unplugged was implemented, which involves diverse activities that teach Computer Science through games and puzzles, cards, strings, colored pencils, and other resources, demonstrating an unplugged approach. Meanwhile, in Singapore, the PlayMaker program was developed, which addresses Computer Science through educational robotics with kits such as BeeBot, Circuit Stickers, and KIBO, integrating technological resources into learning. Finally, in Macedonia, Computer Science is incorporated as a compulsory subject where unplugged activities are included in its development. The author states that "without activities there is no learning, and it is by doing that we learn" (p. 9), which demonstrates that the development of Computer Science does not depend exclusively on the use of digital devices, but rather on how strategies, resources, and methodologies are articulated in the classroom. In this way, approaches such as Montessori and mastery learning reinforce that active learning and participation are essential elements in the construction of the CT.

In addition, Roncoroni and Bailón Maxi (2020) propose an alternative to digital literacy without computers, using resources such as the Turing machine and form grammars, linking computer science with art, creativity, the environment and cultural customs, highlighting the role of the humanities and natural processes as relevant elements in educational processes.

The reviewed studies show that unplugged CT is not only an alternative when technological resources are lacking or limited, but also a way to foster active and contextualized learning experiences that do not require digital devices. These experiences promote skills such as sequencing, collaborative work, and logical and critical thinking, which are essential in various disciplines and real-life situations. Furthermore, this approach demonstrates that CT can be achieved without relying on digital technologies, thus expanding its potential for implementation in diverse educational contexts.

Problem solving

Problem-solving is a fundamental component and a cross-cutting strategy in the development of critical thinking, involving processes such as exploring, representing, and analyzing situations to find solutions. In education, this approach is not limited to the use of specific tools or resources but rather guides how students address problematic situations in an organized manner, fostering the development of logical and analytical thinking in different contexts.

In this regard, Ortega Ruipérez (2018) analyzes the relationship between cognitive thinking and problem-solving through structured activities that guide students in processes such as decomposition, abstraction, and algorithmic thinking, which contributes to the representation and design of solutions. From a complementary perspective, Ortega Ruipérez and Asensio Brouard (2021) address the ability to solve complex problems through cognitive thinking, using the PISA 2012 assessment framework for students aged 14 to 16 as a reference. Taken together, these studies demonstrate that problem-solving is not only a key element in the development of cognitive thinking but also a benchmark for its assessment.

In higher education, Brugés Romero and Camperos Villamizar (2021) analyze how computer programming (CP) influences cognitive skills related to problem-solving methods in students from different engineering programs at the University of Pamplona. They use instruments from the Bebras initiative and the Computer Olympiad Talent Search to evaluate processes such as decomposition, abstraction, pattern identification, algorithm design, evaluation, and logical reasoning. The results show an improvement in these skills, suggesting that CP not only strengthens problem-solving but also fosters the development of higher-order cognitive skills, promoting meaningful learning.

In line with these findings, the systematic review by Huerta Jiménez and Albo (2021) shows that critical thinking is not only linked to programming and robotics, but also to problem-solving skills that extend beyond the fields of computer science and information technology. This suggests that critical thinking is becoming established as a competency applicable to different contexts and areas, most frequently mathematics, physics, history, and music, promoting critical and creative thinking.

From a broader perspective, Vázquez Uscanga et al. (2019)address the incorporation of problem-solving in Latin American educational systems such as Argentina, Chile, Mexico, and Uruguay, highlighting that its implementation constitutes a significant challenge that involves rethinking pedagogical practices and the way educational processes are oriented. In this sense, the authors propose that problem-solving should not be limited to its incorporation as a subject, but rather as a cross-cutting theme articulated with different areas of knowledge, aimed at developing skills that allow for addressing problem-solving from diverse perspectives.

ased on all the above, it is evident that the CT contributes to the development of problem-solving skills and is configured as a transversal axis that encompasses various disciplines, fostering meaningful learning that strengthens students' capacity for innovation and effective problem-solving.

Discussion

The results of the literature review show that CT development has been approached from diverse pedagogical and methodological perspectives, depending on the needs of the context, access to different resources, and learning objectives. In this sense, approaches such as block-based programming, educational robotics, gamification, virtual learning environments, *blended* learning, and offline CT demonstrate that there is no single way to promote CT, but rather various approaches that reflect its constant evolution.

In this scenario, the prevalence of approaches based on block programming and educational robotics is evident, where the former is frequently used for the introduction of computational concepts, since it allows visual representation of elements of abstract logic, while the latter is configured as a dynamic platform that favors critical thinking, iterative design and teamwork (Eguchi, 2014).

In contrast, virtual environments, gamified learning scenarios, and offline PCs expand the ways in which PCs can be developed in the classroom by linking more flexible and context-adaptable dynamics. Virtual environments foster abstract thinking through simulations that strengthen student participation (Kafai & Burke, 2015), while blended learning reflects a trend toward flexible, student-centered pedagogies (Papadakis et al., 2017). Gamification, through challenges, rewards, playful dynamics, and game mechanics, promotes motivation and active participation, and offline activities are consolidating as a viable alternative, since they do not require computers or advanced technological infrastructure to address fundamental concepts in resource-limited environments. These differences reflect the approaches that respond to didactic strategies as well as to the conditions of the educational environment in which they are implemented.

Furthermore, the analyzed literature reveals a trend toward integrating diverse approaches, methodologies, and resources for incorporating and promoting computer thinking in the classroom, where strategies such as robotics, block-based programming, and gamification are articulated within broader proposals. This suggests that teaching this type of thinking should not be limited to isolated practices, but rather to more comprehensive processes that respond to the educational needs of a given environment. In this sense, the development of computer thinking depends on the selected approach and the way in which the learning processes are structured, which confirms the point made by Ortega-Ruipérez (2020), who emphasizes the importance of processes of decomposition, abstraction, and solution design. Similarly, proposals such as Trejos Buriticá (2019) demonstrate the need to methodologically guide its teaching, since it is not simply a matter of implementing tools, but rather of integrating them into the classroom.

However, this diversity reveals some limitations. First, the results of this review must be interpreted considering the diverse nature of the publications in terms of their objectives, context, interventions, educational levels, and other factors that hinder comparisons or quantifications, as well as the generalization of the findings. Similarly, the lack of standardized criteria for evaluating the CT (presumably referring to a specific program or initiative) is evident, coinciding with the points raised by Pérez et al. (2021), who emphasize the need to design contextualized instruments that allow for measurement. Furthermore, this situation is compounded by the dependence on factors such as teacher training and access to resources, which creates gaps in implementation, especially in contexts with technological limitations.

In summary, the reviewed studies show that the integration of CT into the curriculum can be done in various ways, taking into account the suitability of the approach, teacher training and the characteristics of the context, highlighting that there is no single path to promote its development and that each of them must be adjusted to the educational purposes and the particularities of the students.

Conclusions

This review highlights the multidimensional nature of computer literacy and its growing relevance in modern educational practices, as it is recognized as a key competency in teaching and learning processes. Based on a synthesis of the analyzed studies, computer literacy goes beyond coding and programming, encompassing a cognitive and pedagogical shift in how students approach problem-solving, abstraction, and knowledge construction.

In this regard, the results show that there are diverse approaches to its development, among which block-based programming and educational robotics stand out as the most recurrent, demonstrating a trend toward the use of technological resources to promote computer literacy, especially when combined with gamified and experiential learning strategies. In contrast, offline computer literacy emerges as an alternative for its inclusion in environments with limited technological infrastructure, making it accessible and adaptable to different contexts. Similarly, a trend toward the integration of approaches is identified, suggesting that its teaching is strengthened by more comprehensive proposals rather than isolated practices.

However, the incorporation of CT into curricula faces challenges related to teacher training, access to technological resources and a lack of unified criteria for its evaluation, as well as the development of models and approaches that allow for systematic and contextualized implementation.

Consequently, CT should be understood not only as technical competence, but also as a transversal cognitive skill that promotes creative, analytical, and reflective thinking. Its development requires continuous pedagogical innovation, as well as a flexible and inclusive curriculum design that responds to the specific characteristics of each context. In this regard, future studies could delve deeper into the design of assessment instruments and the identification of strategies that allow for their implementation at different educational levels.

References

Alsina, A., & Acosta, Y. (2022). Conectando La Educación Matemática Infantil Y El Pensamiento Computacional: Aprendizaje De Patrones De Repetición Con El Robot Educativo Programable Cubetto®. *Innovaciones Educativas*, *24*(37), 133-148. https://doi.org/10.22458/ie.v24i37.4022

Álvarez Herrero, J. F. (2020). Pensamiento Computacional En Educación Infantil, Más Allá De Los Robots De Suelo. *Education in the Knowledge Society (EKS)*, *21*, 11. https://doi.org/10.14201/eks.22366

Auchstetter, A., Witherspoon, E., Ozuzu, O., Margolin, J., & Friedman, L. B. (2023). *Evaluación De the Pack: Integración De Un Juego De Pensamiento Computacional En Aulas De Secundaria*. https://files.eric.ed.gov/fulltext/ED651278.pdf

Balladares Burgos, J. A., Avilés Salvador, M. R., & Pérez Narváez, H. O. (2016). Del Pensamiento Complejo Al Pensamiento Computacional: Retos Para La Educación Contemporánea. *Sophía*, *2*(21), 143. https://doi.org/10.17163/soph.n21.2016.06

Barr, V., & Stephenson, C. (2011). Bringing Computational Thinking to K-12. *ACM Inroads*, *2*(1), 48-54. https://doi.org/10.1145/1929887.1929905

Basogain-Olabe, X., Olabe-Basogain, M. Á., & Olabe-Basogain, J. C. (2015). Pensamiento Computacional a Través De La Programación: Paradigma De Aprendizaje. *Revista de Educación a Distancia (RED)*(46). https://doi.org/10.6018/red/46/6

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer Science Unplugged: School Students Doing Real Computing without Computers. *New Zealand Journal of Applied Computing and Information Technology*, *13*(1), 20-29. https://purehost.bath.ac.uk/ws/portalfiles/portal/214932627/NZJACIT_Unplugged.pdf

Brennan, K., & Resnick, M. (2012). New Frameworks for Studying and Assessing the Development of Computational Thinking. Proceedings of the 2012 Annual Meeting of the American Educational Research Association, Vancouver.http://scratched.gse.harvard.edu/ct/files/AERA2012.pdf

Brugés Romero, A. R., & Camperos Villamizar, Y. d. P. (2021). Influencia Del Pensamiento Computacional En Las Habilidades Cognitivas Propias De Los Métodos De Resolución De Problemas. *Horizontes pedagógicos*, *23*(2), 57-70. https://doi.org/10.33881/0123-8264.hop-23206

Cabra Páez, M. L., & Ramírez Gamboa, S. A. (2021). 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*, 171-187. https://doi.org/10.15517/revedu.v46i1.44970

Chiazzese, G., Fulantelli, G., Pipitone, V., & Taibi, D. (2018). Engaging Primary School Children in Computational Thinking: Designing and Developing Videogames. *Education in the Knowledge Society (EKS)*, *19*(2), 63-81. https://doi.org/10.14201/eks20181926381

Eguchi, A. (2014). Educational Robotics for Promoting 21st Century Skills. *Journal of Automation, Mobile Robotics and Intelligent Systems*, 5-11. https://doi.org/10.14313/jamris_1-2014/1

Eguíluz, A. (2020). *Análisis Del Desarrollo Del Pensamiento Computacional Con Generalización Mediante Retos De Programación Visual* Universidad de Deusto]. http://hdl.handle.net/20.500.14454/733

Gamito, R., Aristizabal, P., Basasoro, M., & León, I. (2022). El Desarrollo Del Pensamiento Computacional En Educación: Valoración Basada En Una Experiencia Con Scratch. *Innoeduca. International Journal of Technology and Educational Innovation*, *8*(1), 59-74. https://doi.org/10.24310/innoeduca.2022.v8i1.12093

García-Valcárcel-Muñoz-Repiso, A., & Caballero-González, Y.-A. (2019). Robotics to Develop Computational Thinking in Early Childhood Education. *Comunicar*, *27*(59), 63-72. https://doi.org/10.3916/c59-2019-06

García, J. M. (2020). La Expansión Del Pensamiento Computacional En Uruguay. *Revista de Educación a Distancia (RED)*, *20*(63). https://doi.org/10.6018/red.410441

García Rodríguez, A. (2022). Enseñanza De La Programación a Través De Scratch Para El Desarrollo Del Pensamiento Computacional En Educación Básica Secundaria. *Academia y Virtualidad*, *15*(1), 161-182. https://doi.org/10.18359/ravi.5883

Gómez Andrade, A., Gastelú, M. E. R., Barajas, J. A. C., & Torres, G. P. (2022). Proyecto Para Fortalecer El Pensamiento Computacional Mediante Una Estrategia Para Mejorar Las Habilidades De Programación En Estudiantes De Ingenierías En Tics. *DIM: Didáctica, Innovación y Multimedia*(40). https://raco.cat/index.php/DIM/article/view/402722

González González, C. (2020). *Pensamiento Computacional Y Robótica En Educación Infantil: Una Propuesta Metodológica Inclusiva* Universidad de Huelva]. http://hdl.handle.net/10272/19545

Grover, S., & Pea, R. (2013). Computational Thinking in K–12. *Educational Researcher*, *42*(1), 38-43. https://doi.org/10.3102/0013189x12463051

Gutiérrez Rodríguez, C. A. (2018). Una Estrategia Tic Para Fortalecer Las Habilidades De Resolución De Problemas Mediante El Pensamiento Computacional. In *Apropiación, Gestión Y Uso Constructivo Del Conocimiento* (pp. 393-412). Redipe. https://redipe.org/wp-content/uploads/2019/04/Libro-simposio-redipe-cali-2018.pdf#page=393

Huerta Jiménez, C. S., & Albo, M. V. (2021). Pensamiento Computacional Como Una Habilidad Genérica: Una Revisión Sistemática. *Ciencia Latina Revista Científica Multidisciplinar*, *5*(1), 1055-1078. https://doi.org/10.37811/cl_rcm.v5i1.311

Kafai, Y. B., & Burke, Q. (2015). Constructionist Gaming: Understanding the Benefits of Making Games for Learning. *Educational Psychologist*, *50*(4), 313-334. https://doi.org/10.1080/00461520.2015.1124022

Korkmaz, H., & Yıldız, M. (2020). El Efecto Del Pensamiento Computacional En Las Habilidades De Resolución De Problemas Y El Rendimiento Académico De Los Estudiantes: Un Metaanálisis. *Educational Research Review*, *29*, 100296. https://doi.org/10.1016/j.edurev.2020.100296

Laura-Ochoa, L., & Bedregal-Alpaca, N. (2021). Analysis of Programming Environments for the Development of Computational Thinking Skills and Teaching Programming to Beginners. https://www.proquest.com/openview/7504eba5c6fe8f2ba820160b1fda9a0e/1?pq-origsite=gscholar&cbl=1006393

Mallart, J., Sepúlveda, F., & Rajadell, N. (2001). Didactics: Concept, Object, and Purposes. In *General Didactics for Educational Psychologists* (pp. 23-57). https://www.researchgate.net/publication/325120200_Didactica_concepto_objeto_y_finalidades

Mancinas González, A., & Montijo Mendoza, M. F. (2021). Pensamiento Computacional Y Aprendizaje Adaptativo En La Resolución De Problemas Con Fracciones. *EPISTEMUS*, *15*(30). https://doi.org/10.36790/epistemus.v15i30.171

Mantilla Güiza, R. R. (2021). *Propuesta Para El Desarrollo Del Pensamiento Computacional Desde Un Ecosistema Digital. Caso: Escuela Politécnica Vicente Azuero, Colombia* Universidad de las Islas Baleares]. http://hdl.handle.net/11201/158247

Molina Ayuso, Á., Adamuz Povedano, N., & Bracho López, R. (2020). La Resolución De Problemas Basada En El Método De Polya Usando El Pensamiento Computacional Y Scratch Con Estudiantes De Educación Secundaria. *Aula Abierta*, *49*(1), 83-90. https://doi.org/10.17811/rifie.49.1.2020.83-90

Montes-León, H., Hijón- Neira, R., Pérez-Marín, D., & Montes-León, S. R. (2020). Improving Computational Thinking in Secondary Students with Unplugged Tasks. *Education in the Knowledge Society (EKS)*, *21*, 12. https://doi.org/10.14201/eks.23002

Moreno, J., Robles, G., Román, M., & Rodríguez, J. D. (2019). No Es Lo Mismo: Un Análisis De Red De Texto Sobre Definiciones De Pensamiento Computacional Para Estudiar Su Relación Con La Programación Informática. *Revista Interuniversitaria de Investigación en Tecnología Educativa*. https://doi.org/10.6018/riite.397151

Mujica de Statzewitch, L. (2021). Evaluación Del Desarrollo Del Pensamiento Computacional En Estudiantes De Educación Primaria Y Media General. *Areté, Revista Digital del Doctorado en Educación*, *7*(13), 35-56. https://saber.ucv.ve/ojs/index.php/rev_arete/article/view/21327

Ortega-Ruipérez, B. (2020). Pedagogía Del Pensamiento Computacional Desde La Psicología: Un Pensamiento Para Resolver Problemas. *Cuestiones Pedagógicas*, *2*(29), 130-144. https://doi.org/10.12795/cp.2020.i29.v2.10

Ortega-Ruipérez, B., & Asensio Brouard, M. M. (2018). Robótica Diy: Pensamiento Computacional Para Mejorar La Resolución De Problemas. *Revista Latinoamericana de Tecnología Educativa - RELATEC*, *17*(2), 129-143. https://doi.org/10.17398/1695-288x.17.2.129

Ortega Ruipérez, B. (2018). *Pensamiento Computacional Y Resolución De Problemas* Universidad Autónoma de Madrid]. http://hdl.handle.net/10486/683810

Ortega Ruipérez, B., & Asensio Brouard, M. (2021). Evaluar El Pensamiento Computacional Mediante Resolución De Problemas: Validación De Un Instrumento De Evaluación. *Revista Iberoamericana de Evaluación Educativa*, *14*(1), 153-171. https://doi.org/10.15366/riee2021.14.1.009

Papadakis, S., Kalogiannakis, M., & Zaranis, N. (2017). Designing and Creating an Educational App Rubric for Preschool Teachers. *Education and Information Technologies*, *22*(6), 3147-3165. https://doi.org/10.1007/s10639-017-9579-0

Papert, S. (1980). *Mindstorms: Niños, Computadoras E Ideas Poderosas*. Basic Books, Inc. https://doi.org/10.5555/1095592

Pérez, J., Castro, J., & Pedroza, O. (2021). Carencias En La Evaluación Del Pensamiento Computacional. *Revista de Filosofía*, *38*(99), 369-379. https://doi.org/10.5281/zenodo.5651282

Polanco Padrón, N., Ferrer Planchart, S., & Fernández Reina, M. (2020). Aproximación a Una Definición De Pensamiento Computacional. *RIED. Revista Iberoamericana de Educación a Distancia*, *24*(1), 55. https://doi.org/10.5944/ried.24.1.27419

Rico, M. J., & Basogain Olabe, X. (2018). Pensamiento Computacional: Rompiendo Brechas Digitales Y Educativas. *EDMETIC*, *7*(1), 26-42. https://doi.org/10.21071/edmetic.v7i1.10039

Roig-Vila, R., & Moreno-Isac, V. (2020). El Pensamiento Computacional En Educación. Análisis Bibliométrico Y Temático. *Revista de Educación a Distancia (RED)*, *20*(63). https://doi.org/10.6018/red.402621

Rojas López, A. (2019). *Escenarios De Aprendizaje Personalizados a Partir De La Evaluación Del Pensamiento Computacional Para El Aprendizaje De Competencias De Programación Mediante Un Entorno B-Learning Y Gamificación* https://doi.org/10.14201/gredos.140444

Roncoroni, O. U., & Bailón Maxi, J. (2020). Pensamiento Computacional. Alfabetización Digital Sin Computadoras. *Revista ICONO 14. Revista científica de Comunicación y Tecnologías emergentes*, *18*(2), 379-405. https://doi.org/10.7195/ri14.v18i2.1570

Sánchez Vera, M. d. M. (2019). El Pensamiento Computacional En Contextos Educativos: Una Aproximación Desde La Tecnología Educativa. *Research in Education and Learning Innovation Archives*(23), 24-39. https://doi.org/10.7203/realia.23.15635

Sandoval, B., Yánez, A., Suárez, F., & Flor, O. (2020). Scratch as a Tool for Developing Computational Thinking in Basic Education. https://www.proquest.com/openview/8c5bbc92b4525a0fe5c989aa8b2202d1

Sociedad Internacional para la Tecnología en la, E., & Asociación de Profesores de, I. (2011). *Definición Operacional Del Pensamiento Computacional Para La Educación Primaria Y Secundaria*. https://cdn.iste.org/www-root/Computational_Thinking_Operational_Definition_ISTE.pdf

Sotelo Martín, J. A. (2022). Alfabetización Y Pensamiento Computacional En Docentes: Un Estudio Descriptivo. *Prisma Social: Revista de Investigación Social*(38), 137-157. https://dialnet.unirioja.es/servlet/articulo?codigo=8532276&info=resumen&idioma=SPA

Terroba Acha, M., Ribera, J. M., & Lapresa, D. (2021). Computational Thinking in Contextualized Problem Solving in a Story in Early Childhood Education. *Edma 0-6: Educación Matemática en la Infancia*, *9*(2), 73-92. https://doi.org/10.24197/edmain.2.2020.73-92

Terroba Acha, M., Ribera Puchades, J. M., & Lapresa Ajamil, D. (2021). Cultivando El Talento Matemático En Educación Infantil Mediante La Resolución De Problemas Para Favorecer El Desarrollo Del Pensamiento Computacional. *Contextos Educativos. Revista de Educación*(28), 65-85. https://doi.org/10.18172/con.5008

Trejos Buriticá, O. I. (2019). Eps: Metodología Para Resolución De Enunciados En Ciencias Básicas Apoyándose En Pensamiento Computacional. *Revista EIA*, *16*(32), 85-96. https://doi.org/10.24050/reia.v16i32.1266

Triantafyllou, S. A., Sapounidis, T., & Farhaoui, Y. (2024). Students' Development of Computational Thinking and Teachers' Professional Development Via Bebras through Gamification: A Systematic Literature Review. *Data and Metadata*, *3*. https://doi.org/10.56294/dm2024.582

Urquizo, G., Vidal, E., & Castro, E. (2021). Incorporación De Pensamiento Computacional En Ingenierías Como Soporte a La Competencia De Desarrollo De Problemas: Jugando Con Lightbot. *RISTI - Revista Ibérica de Sistemas y Tecnologías de Informática*(E42), 189-197. http://www.risti.xyz/issues/ristie42.pdf

Vázquez Uscanga, E. A., Bottamedi, J., & Brizuela, M. L. (2019). Pensamiento Computacional En El Aula: El Desafío En Los Sistemas Educativos De Latinoamérica. *Revista Interuniversitaria de Investigación en Tecnología Educativa*. https://doi.org/10.6018/riite.397901

Velásquez Montero, B. A. (2022). *Desarrollo Del Pensamiento Computacional En La Primera Infancia* Universidad del Norte]. http://hdl.handle.net/10584/10267

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33–35. https://doi:10.1145/ 1118178.1118215.

Yang, W. (2025). A Three-Phase Professional Development Approach to Improving Robotics Pedagogical Knowledge and Computational Thinking Attitude of Early

Childhood Teachers. *Computers & Education*, *231*, 105282. https://doi.org/10.1016/j.compedu.2025.105282

Yilmaz, T., & İzmirli, S. (2023). Effect of Unplugged and Plugged Coding Activities on Secondary School Students' Computational Thinking Skills. *Journal of Educational Technology and Online Learning*, *6*(4), 1180-1193. https://doi.org/10.31681/jetol.1375335

Zapata-Ros, M. (2015). Pensamiento Computacional: Una Nueva Alfabetización Digital. *Revista de Educación a Distancia (RED)*(46). https://doi.org/10.6018/red/46/4

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

Zhang, X., Chen, Y., Hu, L., Hwang, G.-J., & Tu, Y.-F. (2025). Developing Preschool Children's Computational Thinking and Executive Functions: Unplugged Vs. Robot Programming Activities. *International Journal of STEM Education*, *12*(1). https://doi.org/10.1186/s40594-024-00525-z