Revista ARACÊ
ISSN: 2358-2472

CODING THE FUTURE: PROGRAMMING IN THE FORMATION OF YOUNG MINDS

https://doi.org/10.56238/arev7n2-194

Submitted on: 17/01/2025 **Publication date:** 17/02/2025

Vanessa Morgado Madeira Caldeira[1], Albanita Ferreira Lima[2], Nélia Elisabete Fagundes de Souza[3], Patricia Cardoso Nicolau[4] and Arlindo Moreira de Sousa[5]

ABSTRACT

This study investigated the implementation of programming teaching in schools and its impact on the development of students' skills, with a focus on primary and secondary education. The general objective was to analyze how programming can be integrated into the school curriculum, highlighting its benefits in the development of students' technical, cognitive and socio-emotional skills. The methodology adopted was qualitative, with a bibliographic approach, through the analysis of academic works, articles and other relevant studies. The results indicated that the teaching of programming contributes to the formation of essential skills, such as logical thinking, perseverance, teamwork and autonomy. However, significant challenges were also identified, such as the lack of adequate infrastructure in schools and the need for teacher training. The analysis revealed that, despite the difficulties, the program offers opportunities for digital inclusion, promoting greater equity in access to technological knowledge and preparing students for the challenges of the job market. The final considerations pointed out that, although the benefits of teaching programming are recognized, continuous investments in pedagogical resources and teacher training are necessary. In addition, it was suggested that studies be carried out to expand issues related to teaching methodologies and public policies that encourage the implementation of programming in schools.

Keywords: Programming. Primary education. Skills Development. Digital Inclusion. Labor Market.

[1] Master in Emerging Technologies in Education
MUST University
E-mail: pedagogavanessamorgado@gmail.com
LATTES: https://lattes.cnpq.br/0631539626967045
[2] School Administration Specialist
Vale do Acaraú State University (UVA)
E-mail: albanitalima17@yahoo.com.br
LATTES: https:///lattes.cnpq.br/3385641927372246
[3] Specialist in Dynamics of and In the Classroom
Fluminense Federal University (UFF)
E-mail: pedagoganelia2@gmail.com
LATTES: http://lattes.cnpq.br/7944703648383477
[4] Specialist in Institutional Psychopedagogy
Candido Mendes University (UCAM)
E-mail: patyc.nicolau@gmail.com
LATTES: https://lattes.cnpq.br/3370559036759170
[5] Specialist in Literature
Vale do Acaraú State University (UVA)
E-mail: arlindo.sousa@prof.ce.gov.br
LATTES: http://lattes.cnpq.br/6857946368354404

Revista ARACÊ
ISSN: 2358-2472

INTRODUCTION

Programming has been consolidated as an essential skill in the contemporary world, in which technology permeates almost all spheres of society. It is not only a competence focused on the area of informatics, but a fundamental tool for the development of critical and logical thinking, contributing to the improvement of young people's cognitive skills. The teaching of programming in schools has gained relevance, as new generations are challenged to deal with technological solutions in various contexts. In this scenario, teaching programming emerges as a way to develop in students not only technical knowledge, but also cognitive and socio-emotional skills, which are indispensable for facing the challenges of the digital society.

Therefore, the importance of programming in the formation of young minds is justified, as it is related to the development of skills that transcend the mastery of programming languages, reaching areas such as creativity, problem solving and collaboration. The insertion of programming in the school curriculum not only prepares students for a demanding job market in terms of technological skills, but also contributes to the development of computational thinking, which is recognized as a fundamental competence in the twenty-first century. Although the benefits of teaching programming are discussed, there are still significant challenges in its implementation, especially in terms of school infrastructure, teacher training and adaptation of teaching methodologies. The training of young people for digital reality requires learning codes; it demands a pedagogy capable of integrating programming into the teaching-learning process in a contextualized way.

The question that arises is: how can the teaching of programming be implemented in schools, contributing to the integral formation of young people, considering the challenges and needs of the current educational context? From this question, it seeks to understand the impacts of programming on basic education, its teaching methodologies and the skills it helps to develop, both in the cognitive and social spheres. The objective of this research is to analyze how the teaching of programming can be incorporated into the school curriculum, exploring its pedagogical potential and the challenges faced in its implementation in basic education schools.

The text is structured as follows: in the first part, the theoretical framework that contextualizes the teaching of programming in the educational and technological sphere will be presented. Then, the development will address programming teaching

Revista ARACÊ
ISSN: 2358-2472

methodologies, their application in schools and the impact on the development of cognitive and socio-emotional skills. The methodology used for the elaboration of this research will be detailed, followed by an analysis of the results and pertinent discussions on the subject. Finally, the final considerations will be presented, highlighting the contributions of the study to education and suggesting possible paths for the implementation of the teaching of programming in schools.

THEORETICAL FRAMEWORK

The theoretical framework is organized in order to provide an understanding of the role of programming in the education of young people, addressing fundamental concepts that support the pedagogical practice of this area. At first, the history of programming teaching will be presented, highlighting the evolution of this practice in the educational context and its growing relevance in the twenty-first century. Then, the concept of computational thinking, its application in the learning process, and how it contributes to the development of essential cognitive skills such as logic, creativity, and problem-solving will be discussed. The framework also explores the methodologies of teaching programming, focusing on the tools used in the school environment and the challenges faced by educators in implementing these practices. Finally, the theoretical framework will address the skills that programming helps to develop in students, both in the technical field and in socio-emotional skills, such as teamwork, perseverance and autonomy.

PROGRAMMING IN ELEMENTARY AND SECONDARY EDUCATION

The implementation of programming teaching in schools has gained prominence in recent years, especially in primary and secondary education, due to its relevance for the development of essential skills in young people. According to Bobsin, Nunes and Kologeski (2020), programming not only contributes to technical learning, but also stimulates problem-solving and logical thinking, fundamental skills in the current educational context. In this sense, André (2018, p. 1) highlights that:

> Computational thinking in school favors the revision of one's own productions, always considering them in process. The purpose of the article is to instigate the reader to get in touch with a vision of learning strategy, digital authorship and construction of citizenship that assumes the function of establishing connections with the teacher and student, in a movement that implies understanding the paradigm of computational thinking and the development of activities that prioritize critical thinking and reflection. It also aims to discuss the production of artifacts

Revista ARACÊ
ISSN: 2358-2472

> based on the process of digital authorship, so that students take responsibility for their own learning process, acquiring skills that allow them to continue learning throughout life.

From this perspective, computational thinking, as pointed out by Castilho, Grebogy, and Santos (2019), offers fundamental support for the construction of cognitive and creative skills, indispensable elements for the formation of students prepared for the challenges of the twenty-first century. The term computational thinking introduces a new approach to cognitive science, arguing that the insertion of Computer Science concepts in basic education contributes to the development of a differentiated abstraction skill, applicable to problem solving in various areas of knowledge (Abar; Santos, 2020).

> Computational thinking, as a cognitive process, systematizes the steps of problem solving, the algorithm, which can be applied in other sciences. Thus, the project aims to offer basic school teachers, preferably from public schools, training in the context of computational thinking and to evaluate the impact of these teachers' understanding of concepts of technologies and algorithms. (Abar; Santos, 2020, p. 11).

The inclusion of programming in school curricula also allows young people to become familiar with the logic of the technologies that dominate the contemporary world. In this context, computational thinking should not be seen only as a resource for those who will pursue careers in the technology field, but as an essential tool for the development of logical reasoning and structured problem solving in various areas of knowledge. Thus, basic education should include initiatives that promote the development of computational thinking, ensuring that students acquire the necessary skills to deal with the growing digitalization of society and the labor market.

According to Pereira, Araújo and Bittencourt (2019), when learning to program, students not only acquire technical skills, but also develop skills such as critical thinking and the ability to abstract, which are crucial for adapting to the demands of a technological job market. This approach goes beyond the simple mastery of programming languages, incorporating the idea that programming can be a tool for transformation and innovation in teaching. In this context, André (2018) points out that:

> In recent years, the conceptions about computational thinking have undergone modifications that have allowed this proposal to be brought closer to the student's daily life, that is, to his real world, making it present and concrete. We can see that the concern with computational thinking is not restricted to the school and the formal curriculum of the disciplines. The importance of computational thinking in

Revista ARACÊ
ISSN: 2358-2472

> today's society produces an important pedagogical movement called: science, technology and society. This trend takes into account the current impact of computational thinking, involving an interdisciplinary view that disregards the compartmentalization of knowledge between different areas. (André, 2018, p. 97).

In addition, the implementation of programming teaching in schools contributes to the democratization of access to technologies, expanding learning opportunities and social insertion of students in a digital world. Conforto, Cavedini and Miranda (2018) highlight that learning to code can help reduce educational inequalities, providing students with the possibility of developing digital skills essential for their future, regardless of their socioeconomic context. The inclusion of programming in schools also provides a space for the development of socio-emotional skills, such as perseverance, collaboration, and the ability to deal with challenges, as pointed out by França and Tedesco (2021), who observe the importance of playfulness in the teaching of programming to strengthen these skills.

Therefore, the implementation of programming in schools is an educational strategy for the integral development of young people, by integrating cognitive, digital and socio-emotional skills. Programming should not be seen only as technical content, but as a way to strengthen students' capacities for the digital world and for the formation of critical and active citizens in contemporary society.

METHODOLOGIES AND TOOLS FOR TEACHING PROGRAMMING

The implementation of programming teaching in schools involves the adoption of various pedagogical methodologies and technological tools, which aim to facilitate student learning and make the process accessible. According to Araújo and Silveira (2018), one of the common methodologies is the playful approach, which uses games and interactive activities to teach programming concepts in a fun and engaging way. This is relevant in elementary school, as it stimulates students' creativity and facilitates the understanding of abstract concepts, such as loops and conditionals. This type of approach not only promotes technical learning but also develops cognitive skills, such as logical reasoning and problem-solving, which are essential for academic and professional success.

In addition, several tools have been adopted in programming education, including *Scratch and Python*. *Scratch,* as highlighted by Castilho, Grebogy and Santos (2019), is one of the popular tools for beginners, especially for children and teenagers, due to its simple and intuitive graphical interface. By using code blocks, students can program without the need to type complex codes, which makes learning accessible and less

Revista ARACÊ
ISSN: 2358-2472

intimidating. This visual approach makes it easier to understand the fundamental concepts of programming, allowing students to focus on learning programming logic rather than worrying about code syntax.

On the other hand, *Python*, as highlighted by Bobsin, Nunes, and Kologeski (2020), is a programming language used in high school, due to its simplicity and versatility. Its clear and readable syntax allows students to focus on programming concepts without getting lost in complex details. Python's versatility also makes it an ideal tool for addressing a variety of topics, from creating algorithms to developing simple applications, providing a foundation for students who want to pursue a career in the fields of computer science and engineering.

In addition to the tools mentioned, other approaches, such as unplugged programming, have been suggested to complement programming education. According to Oliveira, Cambraia, and Hinterholz (2021), unplugged programming involves activities that teach programming concepts without the use of computers, such as board games or classroom activities that simulate the coding process. This methodology is especially useful to introduce students to computational thinking before using digital tools, allowing them to understand the underlying logics of programming in an accessible way.

Therefore, the combination of interactive pedagogical methodologies and appropriate technological tools, such as *Scratch and Python*, contributes to the teaching of programming, making it accessible to students. These approaches not only teach the necessary technical skills, but also develop cognitive and socio-emotional competencies essential for the formation of critical citizens prepared for the challenges of the twenty-first century.

THE IMPACT OF TEACHING PROGRAMMING ON THE JOB MARKET AND ON THE TRAINING OF YOUNG PROFESSIONALS

Teaching programming has become an essential tool to prepare young people for the demands of the job market, especially in a demanding digital context. According to Bobsin, Nunes, and Kologeski (2020), programming not only develops technical skills, but also prepares students for the work environment by promoting skills such as problem-solving, critical thinking, and the ability to innovate, characteristics valued in today's job market. Learning to code provides young people with the tools they need to enter fields of technology, computer science, and engineering, professions that have seen a significant

Revista ARACÊ
ISSN: 2358-2472

increase in demand for skilled professionals. This type of training, therefore, not only opens doors in the job market, but also enables rapid adaptation to digital transformations that impact various sectors.

In addition, the inclusion of programming in the school curriculum contributes to the training of professionals capable of dealing with rapid technological changes and new market needs. Castilho, Grebogy and Santos (2019) highlight that mastery of programming allows young people to develop a flexible and adaptable professional profile, essential to deal with constant technological innovations. Programming therefore offers cross-cutting learning that prepares students for a career while strengthening their ability to think logically and communicate in a professional setting. This makes programming a key competency in the development of future tech professionals, as it allows them to understand the tools that drive digital transformation in various industries.

In addition to technical preparation, the program also contributes to the training of young professionals in the socio-emotional aspect. According to Pereira, Araújo and Bittencourt (2019), the challenges faced by students during the programming learning process, such as the need to persist in the face of mistakes and the requirement for analytical thinking, promote the development of important socio-emotional skills, such as resilience, patience and teamwork. These skills are critical to success in the job market, where the ability to work under pressure and adapt to new situations is required. Thus, teaching programming not only trains students in the technical aspect, but also strengthens behavioral and emotional aspects that are decisive for professional success.

Therefore, the program prepares young people for the job market, developing technical and socio-emotional skills essential for their insertion in demanding and technological areas. The training offered by schools, through the teaching of programming, contributes to the training of qualified, resilient and innovative professionals, capable of facing the challenges of the contemporary labor market and contributing to the technological evolution of industries.

METHODOLOGY

The methodology adopted to carry out this research was of a bibliographic nature, characterized as a qualitative study. According to Santana, Narciso and Fernandes (2025, p. 9),

Revista ARACÊ
ISSN: 2358-2472

Bibliographic research, in turn, aims to gather information on a topic from materials already published. This approach is essential to provide theoretical grounding and direct subsequent investigations. Documentary research, on the other hand, differs by working with sources that have not yet been analyzed, such as official documents, photographs, letters, and films, contributing to a new understanding of unexplored materials. Another relevant method is the *ex-post-facto search,* characterized by investigating situations in which independent and dependent variables have already occurred. This approach seeks to understand the cause-and-effect relationships between past events and subsequent phenomena. As pointed out, the methodology is used in academic works, such as monographs and scientific initiation projects, as it offers a differential in the analysis of concrete situations.

Thus, the research approach seeks an analysis of the existing literature on the theme 'programming in the formation of young minds', in order to understand the various theoretical and empirical perspectives related to the teaching of programming in schools. The research was conducted through the analysis of articles, books, dissertations, theses and other relevant academic documents that address both the benefits and challenges of teaching programming in basic education (Santana; Narciso, 2025).

Data collection was carried out by consulting academic databases, such as *Google Scholar*, *Scielo* and journals specialized in the area of education and technology, as well as specific sources on the theme of educational programming. For data analysis, a critical and systematic reading procedure of the selected sources was used, with the objective of identifying and organizing the main theoretical, methodological and practical contributions on the subject. The analysis techniques involved the comparison and synthesis of information, with the construction of a comparative table of the main approaches to teaching programming in basic education, highlighting methodologies, benefits and challenges presented in the reviewed studies.

The following table presents a synthesis of the main sources used in this research, highlighting the authors, titles of the works, year of publication and type of work, organized according to relevance to the theme. This table was prepared in order to provide the reader with an overview of the theoretical contributions consulted, facilitating the understanding of the path taken in the construction of the bibliographic review.

Chart 1: References Used in the Research

Author(s)	Conforming title published	Year	Type of Work
ANDRÉ, C. F.	Computational thinking as a learning strategy, digital authorship and construction of citizenship.	2018	TECCOGS: Digital Journal of Cognitive Technologies, v. 8, n. 1, p. 1-20, São Paulo: Pontifical

Revista ARACÊ
ISSN: 2358-2472

			Catholic University of São Paulo.
ARAÚJO, L.; SILVEIRA, H. U. C	Teaching computational thinking in a public school through a playful platform.	2018	In: Annals of the Workshops of the Brazilian Congress of Informatics in Education (CBIE), 2018. São Paulo: University of São Paulo.
CONFORTO, D.; CAVEDINI, P.; MIRANDA, R..	Computational thinking in basic education: technological interface in the construction of competencies in the twenty-first century.	2018	Revista Brasileira de Informática na Educação, v. 26, n. 1, p. 1-25, Porto Alegre: Universidade de Passo Fundo.
CASTILHO, M.; GREBOGY, E.; SANTOS, I.	Computational thinking in elementary school I.	2019	In: Annals of the Brazilian Symposium on Informatics in Education (SBIE), 2019. São Paulo: University of São Paulo.
PEREIRA, F. T. S. S.; ARAÚJO, L. G.; BITTENCOURT, R.	Computational thinking interventions in basic education through unplugged computing.	2019	In: Annals of the Brazilian Symposium on Informatics in Education (SBIE), 2019. Porto Alegre: SBC.
ABAR, C.; DOS SANTOS, J.	Computational thinking in elementary school in the age of artificial intelligence: where is the teacher?	2020	In: 1st Congress of Artificial Intelligence in Education, 2020. Porto, Portugal: Instituto Politécnico do Porto.
BOBSIN, R. S.; NUNES, N. B.; KOLOGESKI, A. L..	Computational thinking present in the resolution of investigative problems in mathematics in elementary school.	2020	In: Annals of the XXXI Brazilian Symposium on Informatics in Education (SBIE), 2020. Porto Alegre: SBC.
BREDOW, V. H.	Computational thinking in elementary school: a systematic review of the literature.	2020	In: Annals of the International Congress of Education and Technology (CIET): Horizonte, 2020. São Carlos: Universidade Federal de São Carlos.
FRANÇA, R. S.; TEDESCO, P.	Corporeality, playfulness and storytelling in the promotion of computational thinking in school.	2021	In: Annals of the Brazilian Symposium on Computer Education (SBEC), 2021. Porto Alegre: SBC.

Revista ARACÊ
ISSN: 2358-2472

OLIVEIRA, W.; CAMBRAIA, A. C.; HINTERHOLZ, L. T.	Computational thinking through unplugged computing: challenges and possibilities.	2021	In: Annals of the XXIX Workshop on Computer Education (WEI), 2021. Porto Alegre: SBC.

Source: The authors

After the presentation of the table, it is possible to perceive the diversity of the sources used and how they contribute to the understanding of the role of teaching programming in basic education. The analysis of the works present in this table allowed the construction of an information base, providing subsidies for the development of the research.

RESULTS AND DISCUSSION

The word cloud presented below appears as a visual highlight of the frequent and significant terms present in the frame of reference used for the construction of this research. The highlighted terms reflect the main concepts covered in the following topics, in the results and in the discussions. By observing the word cloud, the reader can notice the emphasis on terms such as 'programming', 'computational thinking', 'basic education', 'technology' and 'skills', which are fundamental for understanding the importance of teaching programming in the educational and social context. From this visualization, the focus of the research on how programming contributes to the development of essential competencies in the current scenario becomes clear.

Image 1 - Word Cloud

Source: The authors

Revista ARACÊ
ISSN: 2358-2472

This word cloud not only organizes the concepts covered, but also helps to synthesize the central points of the work, highlighting the areas of greatest relevance for study and discussion. It serves as an initial guide for understanding the topics explored and facilitates the identification of the connections between the various elements that make up the analysis of programming teaching in schools.

CHALLENGES IN THE IMPLEMENTATION OF PROGRAMMING TEACHING IN SCHOOLS

The implementation of programming teaching in schools faces a series of challenges that involve both structural and pedagogical issues. According to Bredow (2020), one of the main obstacles is the lack of adequate infrastructure, such as computers and quality internet access, which hinders the adoption of technologies and tools necessary for teaching programming. This structural limitation is evident in public schools, where financial resources are often insufficient to ensure that all students have access to the appropriate technologies for learning programming. In addition, the scarcity of material resources can affect the quality of teaching, as teachers rely on digital tools to apply interactive teaching methodologies.

Another challenge highlighted by Castilho, Grebogy and Santos (2019) refers to the training of teachers, who often do not have specific preparation to teach programming. Even though the concept of programming is being integrated into the school curriculum, the lack of teacher training limits the effectiveness of this process. These educators often do not have previous training in the areas of technology or computer science, which can result in difficulties in adapting pedagogical content to methodologies for teaching programming. In addition, the lack of knowledge about teaching tools, such as *Scratch or Python*, can compromise the ability of teachers to guide students properly, hindering the development of the desired technical and cognitive skills.

In addition to problems related to teacher training and infrastructure, resistance to the new by some educational institutions also represents a significant obstacle. According to Araújo and Silveira (2018), many schools face difficulties in adapting the curriculum, since programming is seen as extra content, often relegated to the background in relation to other subjects, such as Portuguese and mathematics. This resistance can be attributed to a lack of understanding about the importance of programming for the development of essential skills in the contemporary context, which results in the undervaluation of this area

Revista ARACÊ
ISSN: 2358-2472

in the school curriculum. The implementation of programming teaching, therefore, requires a cultural change within educational institutions, which needs to be accompanied by public policies that encourage the integration of technologies in the teaching-learning process.

Therefore, the introduction of programming in the school curriculum faces complex challenges that involve not only the lack of infrastructure and adequate teacher training, but also the resistance of the educational institutions themselves. Overcoming these obstacles requires a concerted effort among educators, school administrators, and policymakers to ensure that programming is embedded in the educational process, providing students with the skills they need to meet the challenges of the 21st century.

BENEFITS OF TEACHING PROGRAMMING FOR THE DEVELOPMENT OF SOCIO-EMOTIONAL SKILLS

Teaching programming goes beyond the development of technical skills, also promoting the strengthening of socio-emotional skills essential for the integral formation of students. According to França and Tedesco (2021), programming plays a significant role in the development of perseverance, as students face challenges, errors, and difficulties when programming, which forces them to persist and seek alternative solutions to problems that arise. This process of overcoming is fundamental for building resilience, an important skill both in the school environment and in the professional context. By learning to code, students are encouraged to deal with failures constructively, learning from mistakes and improving their approaches, which contributes to the development of a growth mindset.

In addition, teaching programming contributes to the development of skills related to teamwork. As Oliveira, Cambraia, and Hinterholz (2021) point out, many programming activities, especially those based on projects, require students to work collaboratively to solve complex problems, promoting the exchange of ideas and the division of tasks. This collaboration allows students to learn to respect the opinions of others, negotiate solutions, and collaborate effectively, skills that are essential for success in professional environments. The program therefore provides a space for students to practice cooperation, which strengthens their abilities to work as a team and to resolve conflicts constructively.

Autonomy is also an aspect developed through the teaching of programming. As highlighted by Pereira, Araújo, and Bittencourt (2019), programming requires students to

Revista ARACÊ
ISSN: 2358-2472

make independent decisions, plan their own projects, and work in a self-directed manner, which encourages them to develop self-management skills. This autonomous learning process is fundamental for the formation of individuals capable of managing their time, setting goals, and seeking solutions independently. In addition, by learning to program, students become confident in their ability to deal with complex problems and, at the same time, develop a critical attitude towards the solutions they find, always seeking to improve their knowledge.

Therefore, teaching programming not only empowers students with technical skills but also contributes to the development of fundamental socio-emotional skills, such as perseverance, teamwork, and autonomy. These skills are essential for the formation of critical, autonomous and collaborative citizens, who are prepared to face the challenges of the twenty-first century, both in the educational context and in the labor market.

SOCIAL AND CULTURAL IMPACTS OF TEACHING PROGRAMMING

Teaching programming has proven to be a fundamental tool to promote digital inclusion and contribute to the reduction of educational and social inequalities. According to Bredow (2020), by integrating programming into the school curriculum, students have the opportunity to take ownership of digital technologies, which enables them to participate in the digitized society. The inclusion of programming in schools not only prepares young people for the job market, but also provides a chance to democratize access to technologies, allowing students from different socioeconomic backgrounds to acquire the skills necessary for the digital age.

In addition, according to Castilho, Grebogy and Santos (2019), teaching programming helps to reduce the disparity in access to technological knowledge, providing students from areas with fewer resources the opportunity to develop digital skills that would otherwise be out of their reach. This digital inclusion, when inserted into the school curriculum, allows students, regardless of their origin, to compete on equal terms with their peers from advantaged environments. Learning to code thus becomes a tool of equity, allowing all students, regardless of their social background, to acquire an essential skill for the 21st century.

The impact of teaching programming is also reflected in the cultural change it can generate within educational institutions. According to Pereira, Araújo and Bittencourt (2019), the program promotes a culture of innovation and collaboration in schools,

Revista ARACÊ
ISSN: 2358-2472

transforming the way students relate to technology and to each other. By learning to program, students not only become digitally empowered, but also develop a critical understanding of the technological tools they use, which contributes to the development of an active and reflective posture in the face of the digital society.

Therefore, teaching programming plays a key role in digital inclusion and reducing educational and social inequalities. By providing access to technological knowledge and the development of digital skills, it offers young people the opportunity to overcome social and educational barriers, contributing to the construction of an equitable and digitally integrated society.

FINAL CONSIDERATIONS

The final considerations of this research aim to synthesize the main findings and discuss the contributions of the study, in addition to reflecting on the need for further studies to expand the issues addressed. The central objective of the research was to analyze how the teaching of programming can be implemented in schools, considering the benefits that this practice offers to students, especially with regard to the development of technological, cognitive and socio-emotional skills. Throughout the investigation, several aspects were observed that indicate the importance of programming in the training of young people, both for their insertion in the labor market and for their preparation as critical and well-informed citizens in the digital age.

The first relevant finding of the research is that programming, when inserted in the school curriculum, becomes a tool for the development of technical skills, such as mastery of programming languages and understanding of computational thinking. However, in addition to technical skills, teaching programming also contributes to the development of essential socio-emotional skills, such as perseverance, teamwork, and autonomy. Overcoming common challenges when learning to program, such as the need to debug errors and seek alternative solutions, favors the development of resilience and adaptability, which are fundamental for success in today's world. Programming, therefore, is not limited to learning to code, but involves building a mindset that values reflection, persistence, and collaboration.

In addition, the research showed that, although teaching programming is recognized as an essential skill for the future of students, there are significant challenges in its implementation. The lack of infrastructure in schools, combined with the lack of adequate

Revista ARACÊ
ISSN: 2358-2472

training for teachers, makes it difficult to integrate programming into the curriculum. Such obstacles were identified as limiting factors that, although not impossible to overcome, require investments in infrastructure and teacher training to ensure that students can enjoy the benefits of this training. The resistance of some educational institutions, which still see programming as a secondary content, was also pointed out as a factor that hinders the consolidation of programming as an essential curricular component.

With regard to social and cultural impacts, the findings of this research indicate that the inclusion of programming teaching in schools can play a significant role in reducing educational and social inequalities. The program provides an opportunity to democratize access to technological knowledge, allowing students from different socioeconomic backgrounds to develop digital skills that are key to competitiveness in the labor market. In this sense, programming becomes a tool for equity, which contributes to digital inclusion and to the preparation of citizens prepared for the challenges of the digitalized society.

In terms of contributions, the study reaffirms the importance of teaching programming not only for the development of technical skills, but also as a means of promoting inclusive, resilient, and collaborative education. The analysis of the data suggests that, by learning to program, young people not only acquire essential digital skills, but also develop a critical and autonomous posture in the face of the challenges they face in the digital world. The study also highlighted the need for a pedagogical approach that goes beyond the transmission of technical knowledge, incorporating practices that favor the development of socio-emotional skills that complement academic learning.

Although the results obtained are significant, it is possible that further studies are needed to expand some issues that have not yet been explored. The implementation of programming in the school curriculum is a dynamic and multifaceted process, which involves contextual and institutional variables that can influence the results. Thus, future studies could investigate the effectiveness of different programming teaching methodologies, such as the use of specific tools, such as *Scratch and Python*, and the impact of these approaches on student learning. In addition, it would be interesting to explore how public policies can support the implementation of programming teaching in schools, especially in contexts of inequality of resources.

Finally, the research points to the need for an ongoing commitment from educational institutions, government and society at large to ensure that all young people, regardless of their background, have access to an education that includes teaching programming as an

Revista ARACÊ
ISSN: 2358-2472

essential skill for the 21st century. Overcoming the challenges identified in this study and valuing the teaching of programming in schools are fundamental steps to prepare future generations for the challenges and opportunities of the digital world.

Revista ARACÊ
ISSN: 2358-2472

REFERENCES

1. ABAR, C.; SANTOS, J. Pensamento computacional na escola básica na era da inteligência artificial: onde está o professor? In: **1º Congresso de Inteligência Artificial da Educação**, 2020. Porto, Portugal: Instituto Politécnico do Porto, 2020. Recuperado de https://ined.ese.ipp.pt/sites/default/files/2020-12/1%C2%BA%20Congresso%20de%20Intelige%CC%82ncia_ABAR_DOSSANTOS.pdf. Acesso em 09 de fevereiro de 2025.

2. ANDRÉ, C. F. O pensamento computacional como estratégia de aprendizagem, autoria digital e construção da cidadania. TECCOGS: **Revista Digital de Tecnologias Cognitivas**, v. 8, n. 1, p. 1-20, 2018. São Paulo: Pontifícia Universidade Católica de São Paulo, 2018. Recuperado de https://revistas.pucsp.br/teccogs/article/view/48579. inteligência artificial: onde está o professor? In: 1º Congresso de Inteligência Artificial da Educação, 2020. Porto, Portugal: Instituto Politécnico do Porto, 2020. Recuperado de https://ined.ese.ipp.pt/sites/default/files/2020-12/1%C2%BA%20Congresso%20de%20Intelige%CC%82ncia_ABAR_DOSSANTOS.pdf. Acesso em 09 de fevereiro de 2025.

3. ARAÚJO, L.; SILVEIRA, H. U. C. Ensino do pensamento computacional em escola pública por meio de uma plataforma lúdica. In: **Anais dos Workshops do Congresso Brasileiro de Informática na Educação (CBIE)**, 2018. São Paulo: Universidade de São Paulo, 2018. Recuperado de http://milanesa.ime.usp.br/rbie/index.php/wcbie/article/view/8284. inteligência artificial: onde está o professor? In: 1º Congresso de Inteligência Artificial da Educação, 2020. Porto, Portugal: Instituto Politécnico do Porto, 2020. Recuperado de https://ined.ese.ipp.pt/sites/default/files/2020-12/1%C2%BA%20Congresso%20de%20Intelige%CC%82ncia_ABAR_DOSSANTOS.pdf. Acesso em 09 de fevereiro de 2025.

4. BOBSIN, R. S.; NUNES, N. B.; KOLOGESKI, A. L. O pensamento computacional presente na resolução de problemas investigativos de matemática na escola básica. In: **Anais do XXXI Simpósio Brasileiro de Informática na Educação (SBIE)**, 2020. Porto Alegre: SBC, 2020. Recuperado de https://sol.sbc.org.br/index.php/sbie/article/view/12903. Acesso em 09 de fevereiro de 2025.

5. BREDOW, V. H. O pensamento computacional na escola básica: uma revisão sistemática da literatura. In: **Anais do Congresso Internacional de Educação e Tecnologia (CIET)**: Horizonte, 2020. São Carlos: Universidade Federal de São Carlos, 2020. Recuperado de https://ciet.ufscar.br/submissao/index.php/ciet/article/view/586. Acesso em 09 de fevereiro de 2025.

6. CASTILHO, M.; GREBOGY, E.; SANTOS, I. O pensamento computacional no ensino fundamental I. In: **Anais do Simpósio Brasileiro de Informática na Educação (SBIE)**, 2019. São Paulo: Universidade de São Paulo, 2019. Recuperado de http://milanesa.ime.usp.br/rbie/index.php/wie/article/view/8533. Acesso em 09 de fevereiro de 2025.

Revista ARACÊ
ISSN: 2358-2472

7. CONFORTO, D.; CAVEDINI, P.; MIRANDA, R. Pensamento computacional na educação básica: interface tecnológica na construção de competências do século XXI. **Revista Brasileira de Informática na Educação**, v. 26, n. 1, p. 1-25, 2018. Porto Alegre: Universidade de Passo Fundo, 2018. Recuperado de https://seer.upf.br/index.php/rbecm/article/view/8481. Acesso em 09 de fevereiro de 2025.

8. FRANÇA, R. S.; TEDESCO, P. Corporeidade, ludicidade e contação de história na promoção do pensamento computacional na escola. In: **Anais do Simpósio Brasileiro de Educação em Computação (SBEC)**, 2021. Porto Alegre: SBC, 2021. Recuperado dehttps://sol.sbc.org.br/index.php/educomp/article/view/14479. Acesso em 09 de fevereiro de 2025.

9. OLIVEIRA, W.; CAMBRAIA, A. C.; HINTERHOLZ, L. T. Pensamento computacional por meio da computação desplugada: desafios e possibilidades. In: **Anais do XXIX Workshop sobre Educação em Computação (WEI)**, 2021. Porto Alegre: SBC, 2021. Recuperado dehttps://sol.sbc.org.br/index.php/wei/article/view/15938. Acesso em 09 de fevereiro de 2025.

10. PEREIRA, F. T. S. S.; ARAÚJO, L. G.; BITTENCOURT, R. Intervenções de pensamento computacional na educação básica através de computação desplugada. In: **Anais do Simpósio Brasileiro de Informática na Educação (SBIE)**, 2019. Porto Alegre: SBC, 2019.

11. SANTANA, A. C. de A.; NARCISO, R.; FERNANDES, A. B. Explorando as metodologias científicas: tipos de pesquisa, abordagens e aplicações práticas. **Caderno Pedagógico**, v. 22, n. 1, p. e13333, 2025. DOI: 10.54033/cadpedv22n1-130. Disponível em: https://ojs.studiespublicacoes.com.br/ojs/index.php/cadped/article/view/13333. Acesso em: 12 fev. 2025.

12. SANTANA , Aline Canuto de Abreu; NARCISO , Rodi. Pilares da Pesquisa Educacional: Autores e Metodologias Científicas em Destaque. **Aracê**, v. 7, n. 1, p. 1577–1590, 2025. DOI: 10.56238/arev7n1-095. Disponível em: https://periodicos.newsciencepubl.com/arace/article/view/2782. Acesso em: 12 feb. 2025.