International Journal of Innovative Research and Scientific Studies, 9(4) 2026, pages: 40-48

IJIRSS
International Journal of Innovative Research & Scientific Studies

ISSN: 2617-6548

URL: www.ijirss.com

Integrating computational thinking in preschool education: A case study from an internship experience

Ana Isabel Santos[1], Ana Cláudia Loureiro[2*]

[1]*FCSH - Faculty of Social Sciences and Humanities & NICA - Interdisciplinary Center for Children and Adolescents, University of the Azores, 9500-321, Ponta Delgada, Portugal.*
[1]*Interdisciplinary Centre of Social Sciences, CICS.UAc/CICS.NOVA.UAc; Portugal.*
[2]*Polytechnic Institute of Viseu, School of Education, 3504 – 501, Viseu, Portugal.*
[2]*LE@D – Laboratory of Distance Education and eLearning ,1269-001, Lisboa, Portugal.*

Corresponding author: Ana Cláudia Loureiro (*Email: analoureiro@esev.ipv.pt*)

Abstract

The integration of Computational Thinking (CT) in Early Childhood Education has been widely recognized in the last decade due to its importance as a set of fundamental skills for the education of 21st-century citizens. This study examines the pedagogical practices developed by students in the master's Program in Teaching Informatics at the University of the Azores during their teaching internships in preschool and primary education contexts. The study aimed to identify and analyze 32 unplugged and plugged-in activities, designed and implemented across seven classrooms, to promote CT skills in preschool children. A qualitative, interpretative approach was adopted, based on content analysis. Data were obtained from pedagogical practices involving the design and implementation of unplugged and plugged-in activities with preschool children to promote computational thinking. The results highlight the diversity of pedagogical strategies and underscore CT´s potential to develop various skills in children. The results also indicate that, regardless of whether strategies are implemented with or without technology, algorithmic thinking is one of the most promoted skills, closely linked to the development of laterality and spatial orientation. The results demonstrate that Computer Science Education is effective when pedagogical interventions are systematically planned. In addition, the master's program curriculum demonstrates success in integrating theory and practice, enabling the development of contextualized interventions tailored to contemporary educational needs through both remote and in-person strategies.

Keywords: Computational thinking, Early childhood education, Pedagogical practices, Plugged-in and unplugged strategies.

DOI: 10.53894/ijirss.v9i4.11446
Funding: This work was supported by FCT, I.P., the Portuguese national funding agency for science, research, and technology, under the project UID/4647/2023 – Centre of Social Sciences of Universidade Nova de Lisboa.
History: Received: 21 January 2026 / **Revised:** 6 March 2026 / **Accepted:** 11 March 2026 / **Published:** 3 April 2026

Competing Interests: The authors declare that they have no competing interests.
Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.
Transparency: The authors confirm that the manuscript is an honest, accurate, and transparent account of the study; that no vital features of the study have been omitted; and that any discrepancies from the study as planned have been explained. This study followed all ethical practices during writing.
Institutional Review Board Statement: This research was conducted in accordance with Recommendation No. 1/2018 of the University of the Azores' Ethics Committee (Approval code: No. 1/2023).
Publisher: Innovative Research Publishing

1. Introduction

Computational thinking (CT) is an essential area in the educational context nowadays. It is a set of thinking skills fundamental to the development of 21st-century children and young people as full citizens.

Building on Papert [1] and Papert [2] work on constructionist learning and technology, Wing [3] popularized the concept of computational thinking (CT). According to Wing [3] and Wing [4] developing CT skills is as important as being proficient in writing or mathematics because they enhance high-level thinking skills, teaching children to think deductively and solve everyday problems more accurately.

In an educational context, its development involves implementing different pedagogical strategies, which can broadly be categorized as either unplugged (without technology) or plugged-in (with technology, such as tablets, smartphones, robots, and computers).

The use of technology in early childhood education has increased in recent years, as noted by Pollarolo, et al. [5]. This increase is justified by the need to respond to current societal demands and the opportunities that introducing code at an early age provides. Studies by Relkin, et al. [6] and Ching and Hsu [7] and Zhang, et al. [8] demonstrate that using technology, such as educational robots, positively influences the development of computational thinking (CT) skills.

However, since technology is not necessary for developing CT-related thinking skills, some early childhood education studies seem to indicate that unplugged strategies best respond to children's developmental characteristics. These strategies should be preferred when beginning to work with CT at this educational level [9, 10]. Unplugged strategies incorporate physical actions and the manipulation of objects and the body. This allows them to more effectively address children's different learning styles and facilitate the learning of concepts that could initially be seen as abstract [11].

As Wing [3] points out, CT is not restricted to programming languages or technological artifacts such as hardware or software; rather, the elements and techniques that comprise programming can stimulate it when approached in various ways.

Since 2021, CT has been included in the Portuguese primary school curriculum in mathematics as a cross-curricular skill that contributes to mathematical literacy. According to the Essential Learning Outcomes in Mathematics for the First Cycle of Primary Education [12].

> Computational thinking presupposes the development of integrated practices, such as abstraction, decomposition, pattern recognition, algorithm analysis and definition, and debugging and process optimization. These practices are essential to mathematical activities and provide students with the tools necessary to solve problems, particularly those related to programming.

In particular, CT has been in force since 2021 in the Autonomous Region of the Azores as a project under the responsibility of the Regional Government. The project seeks to develop computational thinking in students in the first and second cycles of basic education. The Regional Secretary for Education, Culture, and Sport considers it "unique in the country" due to its complementary implementation in the school curriculum in all regional schools, initially on a voluntary basis. This project involves activities designed by a multidisciplinary team of Azorean professors and implemented in partnership with classroom professors, often without the need for digital devices.

In this context, there has been a growing need to train more capable teachers to meet the demands of this project and others like it, as well as the demands of an increasingly digital and complex society.

In Portugal, the master's degree in Teaching of Informatics is the teacher training program that qualifies professors for systematic, intentional, and pedagogically sustained work in this field. Designed for primary and secondary school teachers, the curriculum at the University of the Azores provides students with the opportunity to complete internships at all levels of education, from preschool to secondary school. This enables future teachers to work in a coordinated and contextualized manner, adapting to the needs of students from the moment they start school until they graduate. This paper will examine the work developed in this training program, particularly the results of the first internship, which focused on preschool education.

As Pollarolo, et al. [5] point out in their literature review, teachers generally have a positive attitude toward technology and programming activities, regardless of their prior knowledge or experience. However, the authors emphasize the need for

context-appropriate training to develop effective teaching methods. This led us to examine the work of students and future teachers to identify good practices and relevant aspects for developing CT skills in early childhood education.

2. Theoretical Framework

2.1. Computational Thinking

Seymour Papert is widely recognized as the pioneer of the term "computational thinking." Although the term was formally coined in his 1980 book, “Mindstorms: Children, Computers, and Powerful Ideas”, he and Cynthia Solomon had already explored the foundations of the concept in their article, "Twenty Things to Do with a Computer" [13]. In this work, Papert and Solomon investigated the use of the LOGO programming language. They argued that computers should be seen as tools for innovating education as opposed to merely maintaining traditional pedagogical practices. Despite Papert's initial contribution, computational thinking was not formally conceptualized until 2006 by Wing [3].

Wing [3] defined computational thinking as an essential skill for the digital age that promotes developing knowledge and skills to solve problems, create systems, and understand human behavior through computer science principles. She emphasized that computational thinking goes beyond programming and involves the ability to think at multiple levels of abstraction. She describes it as a combination of mathematical thinking and engineering, and its solutions can be applied to everyday tasks. According to Wing, computational thinking should be integrated into educational contexts based on the idea that it is transdisciplinary, universal, and useful for everyone.

Since then, numerous publications on the concept of computational thinking have been produced by different scholars, often associating it with coding. For example, Yokuş and Kahramanoglu [14] argue that computational thinking involves solving problems, designing systems, and understanding human behavior using computer science principles. Similarly, Morze, et al. [15] define computational thinking as solving problems through computer science processes and concepts. These authors consider it a crosscutting skill essential for everyone and applicable in different domains. Ansori [16] on the other hand, defines computational thinking as a problem-solving process based on abstraction, analysis, automation, and modeling.

Riley and Hunt [17] argue that the best way to define computational thinking is to characterize it as the way computer scientists think and reason. However, Valente [18] argues that computational thinking is limited by its definition as a problem-solving process aided by digital technology. The author argues that it is necessary to broaden the concept to include personal, environmental, social, emotional, psychological, and ethical dimensions. Haseski, et al. [19] point out the lack of consensus and the variety of definitions in the literature. The authors argue that computational thinking is a social skill that enables active and systematic decision-making with the support of information and communication technologies (ICT) and collaborative approaches in real situations. This allows for the making of ethical decisions and contributes to personal and social development.

The growing number of definitions on the subject, which has increased exponentially since Wing [3] study, has established computational thinking as a key theme in education. It is currently a relevant topic in discussions about the skills that young people should develop during their schooling, and it has received considerable recognition from the scientific and educational communities as a fundamental skill [18, 20-25].

2.2. The promotion of Computational Thinking in an Educational Context

The International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA) define computational thinking as a problem-solving process, identifying it as an essential skill for 21st-century students [26, 27]. In 2011, both organizations presented the fundamentals of computational thinking for primary and secondary schools. These fundamentals covered concepts and practices such as data collection, analysis and presentation; problem decomposition; abstraction; algorithms and procedures; automation; parallelization; and modelling. They also emphasize that computational thinking is a fundamental skill for everyone, not just computer scientists, as it cuts across different disciplines, including the humanities, mathematics and sciences.

From this perspective, Brennan [21] studies argue that computational thinking develops best through programming language activities, which are now more accessible to children and young people. This is thanks to online tools and platforms such as Scratch, Blockly, Code.org and Tynker, which allow users to create programming projects such as games, animations and interactive stories using visual blocks. According to Brennan, the strategies that children use to solve programming problems, and how they direct these strategies in a structured way to achieve their goals through co-coding, develop computational thinking skills and skills in all areas of knowledge. Kafai [28] adds that computational thinking fosters creative thinking, logical reasoning, and collaborative work — fundamental skills in digital society that should be cultivated in educational contexts. Bers, et al. [29] define computational thinking as 'a new literacy for the 21st century', encompassing a broader range of problem-solving concepts than those traditionally explored by computer science. The authors describe computational thinking as an expressive process that promotes new ways of communicating ideas, with coding playing a fundamental role. Similarly, Resnick and Rusk [25] introduce the concept of 'computational fluency', defined as the ability to use computational technologies to communicate ideas effectively and creatively.

Despite its relevance being widely recognized, introducing computational thinking into educational contexts remains challenging [22, 24]. This is particularly evident in light of the demands of future economic and social development. Valente [18] reiterates this, highlighting that the variety of conceptualizations of computational thinking complicates the topic when it is considered in the context of education, particularly in relation to teacher training.

The integration of CT into education also involves understanding its approach. Both unplugged (without computers/digital technology) and plugged-in (using computers, devices, software and digital tools) activities have important roles in this process, albeit with different advantages and limitations.

Unplugged activities are easily accessible and low-cost [30] representing fewer technical barriers for students and teachers Hsu and Liang [31] while facilitating the initial understanding of abstract concepts [11, 30, 32]. They promote cooperation, creativity, motivation, reasoning, and problem-solving skills, but may pose a risk of loss of interest at older ages due to the lack of direct contact with digital tools.

Plugged-in activities bring children and young people closer to real programming tools and languages [7, 31]. They offer immediate feedback and the possibility of debugging errors and are generally more suitable for complex tasks and digital simulations [32]. They also stimulate the development of digital literacy. However, as Aytekin and Topcu [33] point out, these activities can present challenges in terms of coding skills and technology use. This can distract children during problem solving and requires the use of appropriate technological resources and infrastructure, ultimately influencing their implementation [30].

In order to make informed decisions about the most effective ways to develop computational thinking (CT) in an educational context, it is necessary to consider the advantages and disadvantages of different approaches. The implementation of initiatives to integrate CT into the school environment requires greater training in the development of teachers' digital skills. It is in this context that students on the Informatics Teaching Internship I course were able to develop pedagogical practices aimed at promoting learning strategies focused on developing computational thinking among early childhood education children. This experience will be analyzed in the following sections.

3. Method

3.1. Research

This study takes an interpretative, qualitative methodological approach based on content analysis [34]. Data were obtained from pedagogical practice developed by master's students, involving designing and implementing unplugged and plugged-in activities with preschools children to promote computational thinking.

3.2. Study Context and Participants

The study was conducted as part of the Internship in Informatics Education I curricular unit, which is part of the master's degree in Teaching Informatics at the University of the Azores in Portugal during the 2022–23 academic year. Seven first-year master's students enrolled in the programme participated; all had degrees in Informatics or a related field (e.g. Computer Engineering). Only one participant had a master's degree in Computer Science. The participants' ages ranged from 26 to 50, and at the time, only one had no previous teaching experience in an educational context.

The internship adopted a reflective and experimental methodology involving intervention in educational contexts in pre-school and primary education, totaling 110 contact hours. The internship activities included observing educational practices and teaching in real situations. In total, the internship involved seven classrooms and the implementation of 32 activities.

For data protection reasons, the analysis of activities was carried out by the teachers who wrote this article and monitored the entire design and implementation process. This procedure was necessary because the material contained confidential and sensitive information, including elements that could be used to identify students, schools, and participating children, as well as images obtained during the intervention.

3.3. Data Collection Procedures

The data for this study were collected from the observation grid of activities implemented, as well as from the children's participation in these activities, during the first term of the school year (see Figure 1). Each planned activity was recorded individually according to the five categories determined in the grid: (i) Activities and Re-sources, (ii) PC Objectives and Skills, (iii) Typology, (iv) Observations, and (v) Type of Work. These data were then transcribed into digital portfolios designed by the students themselves, in order to present reports on educational practice observations and lessons learned from activities implemented in real contexts. The portfolios also noted the children's performance during implementation and their main difficulties in mobilizing PC skills or those associated with them [35].

UNIVERSITY OF THE AZORES
Faculty of Social Sciences and Humanities

Master's degree in teaching of informatics

Internship in Informatics Education I

Observation Grid

Activities/resources	CT Objectives and Competencies	Typology	Observations	Type of work
Name and brief description of the activity and resources used	Present the objectives of the activity and the CT skills to be developed.	Refer Plugged-in or unplugged	Critical and reflective description of children's performance (their participation and their main difficulties in mobilizing CT skills or those associated with it).	Specify Individual or in pairs

Figure 1.
Observation grid.

Descriptive statistics were used to analyze the data, including calculating frequencies, percentages and averages of the types of activities and PC skills promoted in each one. The statistical software packages SPSS v25 and Microsoft Excel 365 were used to organize and interpret the results, enabling a systematic and rigorous analysis of the activities. A total of thirty-two activities, designed either individually or in pairs, were analyzed and implemented in the seven activity classrooms to promote PC skills.

4. Results

The data were analyzed in line with the objective of this study, which was to identify and analyze unplugged and plugged-in activities designed and implemented by students with pre-school children to promote computational thinking. These results will be presented in the following section.

The results show that, in terms of type, unplugged activities were more numerous (59.4%; n = 19) than plugged-in activities (40.6%; n = 13).

The analysis carried out found significant differences in the trainees' practices in the activity rooms, depending on the type of activity implemented. The overall average for unplugged activities was 2.71 and for plugged-in activities it was 1.86 (see Table 1). However, when rooms that did not develop one of the activity types were excluded (frequency 0 for plugged-in activities in room 6 and for unplugged activities in room 7), the averages changed to 3.17 and 2.17 respectively, revealing that these activities occurred with greater intensity than the overall average in the rooms in which they were carried out.

Table 1.
Typology of the implemented activities.

Classrooms	Typology			
	Unplugged (n= 19)		Plugged-in (n= 13)	
	F	%	F	%
1	4	21.05	2	15.38
2	4	21.05	2	15.38
3	4	21.05	1	7.69
4	2	10.53	1	7.69
5	2	10.53	4	30.77
6	3	15.79	0	0.0
7	0	0.0	3	23.08
Mean		2.71		1.86

Although the unplugged approach is more prevalent in the sample, these results suggest considerable variation in its adoption across different classrooms.

Class profile, familiarity with technological resources and teaching and learning methodologies, and teaching experience may have influenced the choice of this activity. It is important to note that most students (6) had previous teaching experience at the time; however, this was not in the context of preschool education. This may have influenced the students' choice of activity, since working with plugged-in activities required the use of technological resources with age groups with which the students had no prior teaching experience. To ensure a more in-depth interpretation of the data, the strategies and technological resources used were analyzed to identify any significant differences within the sample that could corroborate our study (Table 2).

Table 2.
Strategies and technological resources used in the implemented activities.

Classrooms	Strategies/ Resources	Typology
1	Strategy: stories, games, and problem solving (4 activities) Resource: Super Doc robot (2 activities)	Unplugged Plugged-in
2	Strategies: games and images (4 activities) Resource: Bee-Bot robot (2 activities)	Unplugged Plugged-in
3	Strategy: stories, videos, and problem solving (4 activities) Resource: Botley robot (1 activity)	Unplugged Plugged-in
4	Strategy: story, role play, and problem solving (2 activities) Resource: Botley robot (1 activity)	Unplugged Plugged-in
5	Strategy: storytelling and problem solving (2 activities) Resource: ScratchJr (4 activities)	Unplugged Plugged-in
6	Strategy: games, videos, and problem solving (3 activities) Resource: objects, and Integrating Computational Thinking in Preschool Education (3)	Unplugged
7	Strategy: problem solving (3 activities) Resouce: ScratchJr e Code.org (3 activities)	Plugged-in

When the strategies and technological resources used were organized, it was found that most students followed some fundamental strategies, such as introducing activities using stories or videos (n = 5), actively exploring problem situations (n = 5), and using games to explore space and laterality (n = 3) prior to the presentation and exploration of educational robots.

These strategies were present in unplugged activities in six of the seven classrooms, with a frequency ranging from two to four activities per classroom. This indicates a predominance of this strategic approach, which is common in preschool educational contexts.

In terms of the resources used for plugged-in activities, educational robots (Super Doc, Bee-Bot and Botley) and the digital platforms ScratchJr and Code.org were used in six of the seven classrooms. However, the frequency with which these resources were used varied considerably, ranging from one to four activities per classroom. Classroom 5 implemented plugged-in activities the most frequently (n = 4), followed by classroom 7 (n = 3) and then classroom 6 (n = 0).

These trends were discussed with the students during the portfolio defense, which was the final part of the internship. As these students had no problems with digital skills, many of the difficulties in implementing plugged-in activities with preschool children were due to the educators' lack of familiarity with the pedagogical issues specific to the field of Education [35].

Once we had analyzed the data on the types of activities and the technological strategies and resources used, we moved on to analyzing the computational thinking skills that these activities were intended to promote. To achieve this, we organized the collected data into five computational thinking (CT) skills: algorithmics, pattern recognition, abstraction, problem solving and decomposition. The results obtained reflect remarkable diversity in the activities' promotion of skill development (see Table 3). Students did not limit themselves to developing only one skill; rather, they attempted to diversify and associate it with other skills, whether computational thinking or not.

Table 3.
CT skills developed in the implemented activities.

CT Skills	Frequency and mean of skills				
	Unplugged	Plugged-in	Total for activity (n=32)		Mean for classroom (n= 7)
	F	F	F	%	
Algorithmics	15	11	26	81.25	3.71
Pattern recognition	2	6	8	25.00	1.14
Abstraction	4	1	5	15.62	0.71
Problem solving	7	4	11	34.37	1.57
Decomposition	0	3	3	9.37	0.43

The promotion of algorithmic thinking, or 'algorithmic', was the most prevalent theme, accounting for 81.25% of all activities (n = 32) and an average of 3.71 occurrences per classroom (n = 7). This indicates that the ability to structure

logical sequences was prioritized throughout the planning and execution of pedagogical proposals, suggesting that students are more familiar with activities that promote this skill and that teachers are more familiar with didactic strategies.

The skills of 'problem solving' (34.37%, mean = 1.57) and 'pattern recognition' (25.00%, mean = 1.14) were moderately frequent, being present in a significant proportion of activities, albeit with lower intensity. The competencies of 'abstraction' (15.62%, mean = 0.71) and 'decomposition' (9.37%, mean = 0.43) were the least prevalent, suggesting that approaches requiring the generalization of concepts or the refinement of complex problems were implemented on a case-by-case basis.

From a global perspective, these results demonstrate a preference for more immediate and procedural application skills at the expense of the more abstract and cognitive structuring dimensions of PC.

In addition to PC skills, analysis of the implemented activities reveals other skills and curricular areas worked on by the trainees in a coordinated manner (Figure 2). Spatial orientation and laterality stand out as two essential skills strongly linked to PC activities, particularly unplugged activities involving concrete objects or children's own bodies.

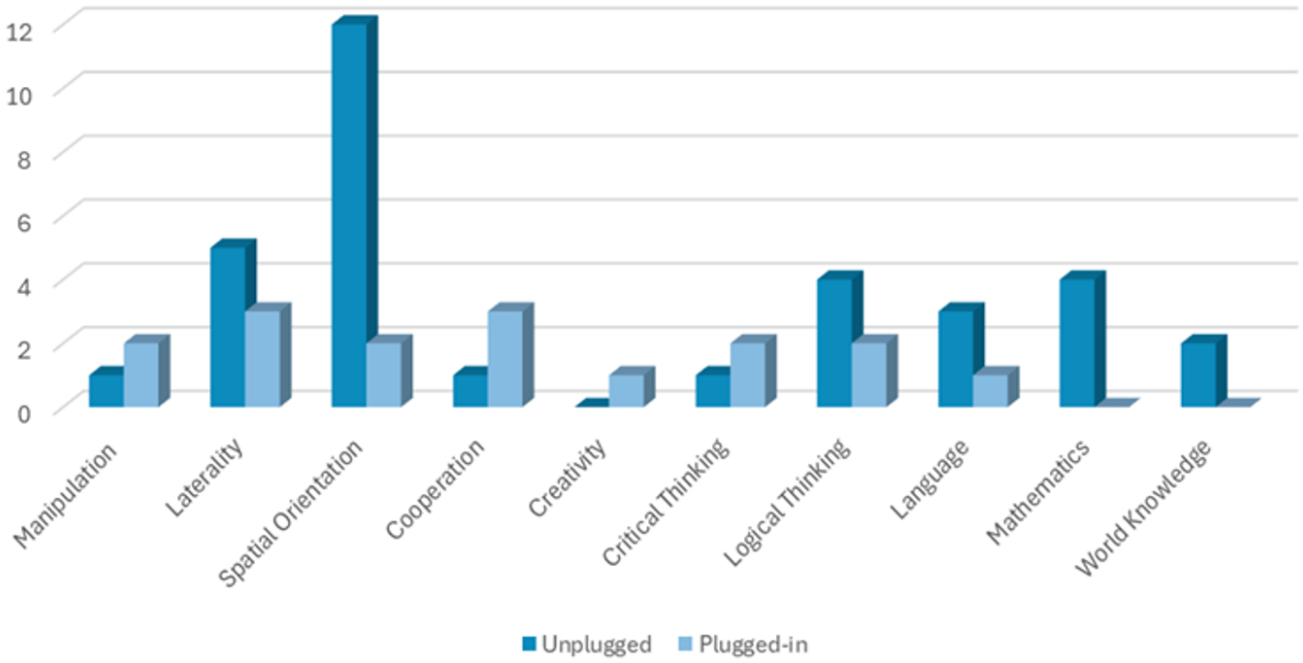

Figure 2.
Other competencies and curricular areas promoted.

5. Discussion

In this study, seven future teachers undergoing pedagogical training to teach informatics in primary and secondary education carried out the curricular unit Internship in Teaching Informatics I, which took place in preschool and 1st cycle primary education contexts in Portugal, designed and implemented unplugged and plugged activities with preschool children to promote computational thinking.

The students' understanding of computational thinking as a set of fundamental 21st-century thinking skills and that CT is a problem-solving process that does not always require the use of computers explains why so many activities designed by students favored unplugged approaches (59.4%; n=19) over plugged activities (40.6%; n=13). This option is in line with the recommendation to use unplugged strategies adapted to children's developmental characteristics as an initial method for introducing computational thinking, which is consistently found in the literature [9, 10]. Furthermore, as suggested in more recent research [11, 30, 32] unplugged strategies, by incorporating physical actions and the manipulation of objects and the body, more effectively address children's different learning styles and allow them to explore fundamental concepts in a concrete, playful and accessible way, without the direct mediation of digital devices, facilitating attention and learning of concepts that, at first glance, could be seen as more abstract.

Another possible interpretation of this result, based on reports of educational practices developed in digital portfolios, is that students on this master's programme have gained a deeper understanding of how preschool-aged children learn. This understanding reinforces the preference for a greater number of unplugged activities and can also be considered one of the factors influencing the choice of strategies and technological resources used in these activities. Using stories or videos to introduce activities, actively exploring problem situations and using games to explore spatial and laterality concepts (strategies present in six out of seven unplugged activities) play an essential pedagogical role in preschool education, particularly when preceding the presentation of more abstract concepts and the manipulation of technological resources such as robots. This is because these strategies are considered more appropriate for the needs and characteristics of this level of education [11, 30, 32].

Plugged-in activities showed an inconsistent implementation pattern across classrooms, with frequencies ranging from 0 to 4 activities. This reveals a tendency to explore technological resources in classrooms 5 and 7 (Table 1), as well as a

tendency not to use them in classroom 6 (Table 1). These results suggest that even though these students have adequate digital skills, they are struggling to apply them to pedagogical practices appropriate to the preschool context. This confirms the gap between technical and pedagogical mastery identified in previous studies by Loureiro and Santos [35]. With regard to CT skills, algorithmic thinking predominated, being present in 81.25% of the implemented activities, indicating that students were more familiar with teaching strategies focused on structuring logical sequences. In contrast, more complex cognitive skills such as 'abstraction' and 'decomposition' were less frequent. This suggests a tendency to prioritize procedural and immediate application over more abstract and structured approaches to CT. There was also a notable articulation with other skills and curricular areas, particularly spatial orientation and laterality. These skills were frequently used in unplugged activities as they involved movement and the use of physical objects, thereby contributing to the consolidation of fundamental learning in conjunction with CT.

6. Conclusions, Limitations and Future Research

6.1. Conclusions

It is a fundamental task for all components of teacher training, both formal and in-formal, to prepare teachers to adopt teaching practices that incorporate aspects relevant to the development of computational thinking from early childhood onwards. In the Autonomous Region of the Azores, the inclusion of CT in the school curriculum for the first and second cycles of basic education in all regional schools since 2021 has made pedagogical training in the use of technologies for future teachers crucial. Consistent with previous studies [5, 35, 36] the findings of this study demonstrate that training in informatics education can be effective when carefully planned and targeted. The curriculum of the master's degree in Teaching of Informatics is also effective, as it enables students to undertake internships where they can apply the theory they have studied in an educational context. This allows them to intervene in a coordinated, contextualized manner that is better adapted to current educational characteristics and needs. This can be justified by the results of this study, which revealed the successful implementation of a variety of unplugged and plugged-in intervention strategies in classrooms.

6.2. Limitations and Future Research

The study has limitations that should be considered when interpreting the results and conducting future research of this nature. In particular, these limitations are related to the training context in which the students found themselves and the processes involved in reconstructing their learning and experiences in the field. The limited number of students on placement also reduces the number of pedagogical situations that can be analyzed. Some of these situations were limited to following a particular approach, as we have seen. In terms of future research, in addition to broadening the analysis of students' pedagogical practices, it would also be beneficial to analyze the impact of these practices on the children they work with. This would help us to understand the extent to which the objectives and skills sought to be enhanced really represent the greatest developmental benefit for children.

References

[1] S. A. Papert, *Mindstorms: Children, computers, and powerful ideas*. New York: Basic Books, 1993.

[2] S. A. Papert, "Keynote lecture', presented at the Keynote at ICMI 17 Conference in Hanoi," 2006. http://dailypapert.com/wp-content/uploads/2012/05/Seymour-Vietnam-Talk-2006.pdf

[3] J. M. Wing, "Computational thinking," *Communications of the ACM*, vol. 49, no. 3, pp. 33-35, 2006. https://doi.org/10.1145/1118178.1118215

[4] J. Wing, "Research notebook: Computational thinking—What and why," *The Link Magazine*, vol. 6, no. 1, pp. 20-23, 2011.

[5] E. Pollarolo, S. Papavlasopoulou, F. Granone, and E. Reikerås, "Play with coding toys in early childhood education and care: Teachers' pedagogical strategies, views and impact on children's development a systematic literature review," *Entertainment Computing*, vol. 50, p. 100637, 2024. https://doi.org/10.1016/j.entcom.2024.100637

[6] E. Relkin, L. E. de Ruiter, and M. U. Bers, "Learning to code and the acquisition of computational thinking by young children," *Computers & Education*, vol. 169, p. 104222, 2021. https://doi.org/10.1016/j.compedu.2021.104222

[7] Y.-H. Ching and Y.-C. Hsu, "Educational robotics for developing computational thinking in young learners: A systematic review," *TechTrends*, vol. 68, no. 3, pp. 423-434, 2024. https://doi.org/10.1007/s11528-023-00841-1

[8] X. Zhang, Y. Chen, L. Hu, G.-J. Hwang, and Y.-F. Tu, "Developing preschool children's computational thinking and executive functions: Unplugged vs. robot programming activities," *International Journal of STEM Education*, vol. 12, no. 1, p. 10, 2025. https://doi.org/10.1186/s40594-024-00525-z

[9] A. Otterborn, K. J. Schönborn, and M. Hultén, "Investigating preschool educators' implementation of computer programming in their teaching practice," *Early Childhood Education Journal*, vol. 48, no. 3, pp. 253-262, 2020. https://doi.org/10.1007/s10643-019-00976-y

[10] A. Saxena, C. K. Lo, K. F. Hew, and G. K. W. Wong, "Designing unplugged and plugged activities to cultivate computational thinking: An exploratory study in early childhood education," *The Asia-Pacific Education Researcher*, vol. 29, no. 1, pp. 55-66, 2020. https://doi.org/10.1007/s40299-019-00478-w

[11] Y. Zeng, W. Yang, and A. Bautista, "Teaching programming and computational thinking in early childhood education: a case study of content knowledge and pedagogical knowledge," *Frontiers in Psychology*, vol. 14, p. 1252718, 2023. https://doi.org/10.3389/fpsyg.2023.1252718

[12] Ministério da Educação – Direção-Geral da Educação (DGE), *Essential math learning objectives – 1st cycle of basic education*. Lisboa: DGE. Disponível em, 2021.

[13] S. Papert and C. Solomon, "Twenty things to do with a computer (AI Memo No. 248). Cambridge, MA: Massachusetts institute of technology, artificial intelligence laboratory. Disponível em: ," 1971. https://dspace.mit.edu/handle/1721.1/5836

[14] E. Yokuş and R. Kahramanoglu, "An overview of computational thinking," *Anemon Muş Alparslan Üniversitesi Sosyal Bilimler Dergisi*, vol. 10, no. 1, pp. 157-173, 2022. https://doi.org/10.18506/anemon.1033403

[15] N. V. Morze, O. V. Barna, and M. A. Boiko, "The importance of computational thinking training for primary school teachers," *Educational Dimension*, vol. 6, pp. 22-39, 2022. https://doi.org/10.31812/educdim.4466
[16] M. Ansori, "Computational thinking in problem solving," *Dirasah: Jurnal Studi Ilmu Dan Manajemen Pendidikan Islam*, vol. 3, no. 1, pp. 111-126, 2020.
[17] D. Riley and K. A. Hunt, *Computational thinking for the modern problem solver*, 1st ed. Boca Raton, FL: Chapman and Hall/CRC, 2014.
[18] J. A. Valente, "Computational thinking, computational literacy, or digital competence? New challenges in education," *Revista educação e cultura contemporânea*, vol. 16, no. 43, pp. 147-168, 2019. https://doi.org/10.5935/reeduc.v16i43.5852
[19] H. İ. Haseski, U. Ilic, and U. Tugtekin, "Defining a new 21st century skill-computational thinking: Concepts and trends," *International Education Studies*, vol. 11, no. 4, pp. 29-42, 2018.
[20] M. U. Bers, "Coding and computational thinking in early childhood: The impact of ScratchJr in Europe," *European Journal of STEM Education*, vol. 3, no. 3, p. 8, 2018. https://doi.org/10.20897/ejsteme/3868
[21] K. Brennan, "How kids manage self-directed programming projects: Strategies and structures," *Journal of the Learning Sciences*, vol. 30, no. 4-5, pp. 576-610, 2021. https://doi.org/10.1080/10508406.2021.1936531
[22] Y. B. Kafai and Q. Burke, *Computational participation: Teaching kids to create and connect through code, in Emerging Research, Practice, and Policy on Computational Thinking, P. J. Rich and C. B. Hodges, Eds*. Cham: Springer International Publishing, 2017.
[23] K. Murcia, E. Cross, and G. Lowe, "Young children's computational thinking: Educator pedagogy fostering children's play and learning with a tangible coding device," *The Australian Educational Researcher*, vol. 52, no. 2, pp. 1261-1279, 2025. https://doi.org/10.1007/s13384-024-00762-9
[24] J. L. Ramos and R. G. Espadeiro, *Introduction to programming in the 1st cycle of basic education*. Lisboa, Portugal: Evaluation studies. DGE – Direção Geral de Educação, 2016.
[25] M. Resnick and N. Rusk, "Coding at a crossroads," *Communications of the ACM*, vol. 63, no. 11, pp. 120-127, 2020. https://doi.org/10.1145/3375546
[26] CSTA, "CSTA," 2022. http://www.csteachers.org
[27] ISTE, "'Computational Thinking Leadership Toolkit, first edition'. International Society for Technology in Education," 2011. https://cdn.iste.org/www-root/2020-10/ISTE_CT_Leadership_Toolkit_booklet.pdf
[28] Y. B. Kafai, "From computational thinking to computational participation in K--12 education," *Communications of the ACM*, vol. 59, no. 8, pp. 26-27, 2016. https://doi.org/10.1145/2955114
[29] M. U. Bers, C. González-González, and M. B. Armas–Torres, "Coding as a playground: Promoting positive learning experiences in childhood classrooms," *Computers & Education*, vol. 138, pp. 130-145, 2019. https://doi.org/10.1016/j.compedu.2019.04.013
[30] P. Chen, D. Yang, A. H. S. Metwally, J. Lavonen, and X. Wang, "Fostering computational thinking through unplugged activities: A systematic literature review and meta-analysis," *International Journal of STEM Education*, vol. 10, no. 1, p. 47, 2023. https://doi.org/10.1186/s40594-023-00434-7
[31] T.-C. Hsu and Y.-S. Liang, "Simultaneously improving computational thinking and foreign language learning: Interdisciplinary media with plugged and unplugged approaches," *Journal of Educational Computing Research*, vol. 59, no. 6, pp. 1184-1207, 2021. https://doi.org/10.1177/0735633121992481
[32] E. Polat and R. M. Yilmaz, "Unplugged versus plugged-in: Examining basic programming achievement and computational thinking of 6th-grade students," *Education and Information Technologies*, vol. 27, no. 7, pp. 9145-9179, 2022. https://doi.org/10.1007/s10639-022-10992-y
[33] A. Aytekin and M. S. Topcu, "Improving 6th grade students' creative problem solving skills through plugged and unplugged computational thinking approaches," *Journal of Science Education and Technology*, vol. 33, no. 6, pp. 867-891, 2024. https://doi.org/10.1007/s10956-024-10130-y
[34] L. Bardin, *Análise de conteúdo* (Edições). Lisboa: Edições, 2016.
[35] A. C. Loureiro and A. I. Santos, "The portfolio as a tool for learning and assessment in the Internship in Teaching Informatics," in *2023 International Symposium on Computers in Education (SIIE) (pp. 1-6). IEEE*, 2023.
[36] A. C. Loureiro and A. I. Santos, "The pedagogical training of future ICT teachers: An experience report from the University of the Azores," 2023. https://iated.org/concrete3/view_abstract.php?paper_id=105045