Sociedade Brasileira de Educação Matemática

Computational Thinking with Python in the Mathematics Education: Interdisciplinary Connections and Challenges in the New High School

Alireza Mohebi Ashtiani
Universidade Tecnológica Federal do Paraná
Londrina, PR — Brasil
✉ ashtiani@utfpr.edu.br
0000-0003-2579-3759

Tatielen Demarchi
Secretaria de Educação do Estado de São Paulo
Ourinhos, SP — Brasil
✉ atielen@alunos.utfpr.edu.br
0000-0003-3421-0172

2238-0345

10.37001/ripem.v15i2.4463

Received • 26/01/2025
Approved • 20/04/2025
Published • 28/05/2025

Editor • Gilberto Januario

Abstract: A number of innovative strategies have been proposed to stimulate logical thinking, most notably the integration of Computational Thinking into the National Common Curriculum Base. This approach, especially in the context of the New High School, promotes the use of technologies such as computers and programming languages to complement traditional teaching practices. This study analyzes the impact of integrating Computational Thinking and Mathematics in the context of the New High School, using a Formative Itinerary as a methodological strategy. The aim is to systematically assess how the inclusion of playful and technological resources can contribute to making learning more meaningful and engaging. This analysis is based on theoretical principles and practical evidence.

Keywords: Mathematics Education. Computational Thinking. New High School. Interdisciplinarity.

Pensamiento Computacional con Python en la Educación Matemática: Conexiones Interdisciplinarias y Desafíos en el Nuevo Currículo de Bachillerato

Resumen: Se han desarrollado diversas estrategias para ayudar a los estudiantes a mejorar sus habilidades de razonamiento lógico, incluyendo la integración del Pensamiento Computacional en la Base Nacional Común Curricular. Este enfoque, especialmente en el contexto del Nuevo Currículo de Bachillerato, promueve el uso de tecnologías como computadoras y lenguajes de programación para complementar las prácticas de enseñanza tradicionales. Este estudio analiza el impacto de la integración del Pensamiento Computacional y las Matemáticas dentro del marco del Nuevo Bachillerato, utilizando un Itinerario Formativo como enfoque metodológico. El objetivo es evaluar de manera sistemática cómo la inclusión de recursos lúdicos y tecnológicos puede contribuir a hacer el aprendizaje más significativo y atractivo. Este análisis se fundamenta en principios teóricos y evidencias prácticas.

Palabras clave: Educación Matemática. Pensamiento Computacional. Nuevo Bachillerato. Interdisciplinariedad.

Pensamento Computacional com Python no Ensino da Matemática: Conexões Interdisciplinares e Desafios no Novo Ensino Médio

Resumo: Diversas estratégias inovadoras têm sido propostas para estimular o raciocínio lógico, com destaque para a integração do Pensamento Computacional na Base Nacional Comum Curricular. Essa abordagem, especialmente no contexto do Novo Ensino Médio, promove o uso de tecnologias, como computadores e linguagens de programação para complementar práticas de ensino tradicional. Este estudo analisa o impacto da integração entre Pensamento Computacional e Matemática no contexto do Novo Ensino Médio, utilizando um Itinerário Formativo como estratégia metodológica. O objetivo é avaliar, de maneira sistemática, como a inclusão de recursos lúdicos e tecnológicos pode contribuir para tornar o aprendizado mais significativo e engajador. Essa análise fundamenta-se em princípios teóricos e evidências práticas.

Palavras-chave: Educação Matemática. Pensamento Computacional. Novo Ensino Médio. Interdisciplinaridade.

1 Introduction

In recent decades, there has been much discussion about students' performance in mathematics and the skills and competencies they develop in the classroom. Masola and Allevato (2019) point out that learning difficulties, not just in mathematics, but in many areas, can be influenced by a variety of factors, such as organic, emotional and social issues. Early identification of these difficulties is fundamental to effectively supporting the teaching-learning process. Considering this, the role of the teacher, as a mediator of knowledge and facilitator of the integration of students' experiences, becomes increasingly challenging, transforming these elements into formalized experiences. This requires the construction of solid structures and well-developed teaching strategies capable of transforming the diversity present in classrooms into opportunities for the development of meaningful skills.

Educators, teachers and researchers in the field of Mathematics Teaching point out that the lack of appropriate methodologies and didactics that take into account the diversity present in classrooms is a determining factor in the low performance of students. In addition, they point to the imminent need to reformulate the curriculum, especially in secondary education, since traditional approaches and practices have proved insufficient to promote effective and meaningful learning.

The implementation of the New High School represents an important transformation in the Brazilian education system, seeking to align education with the demands and challenges of the 21st century and the needs of a constantly evolving society. Among the most important changes are the reorganization of the curriculum around areas of knowledge, replacing the model based on isolated subjects; and the offer of training itineraries that allow students to deepen their specific knowledge or opt for technical and vocational training. This reform includes the offer of different educational itineraries, allowing students to deepen their knowledge in areas such as Languages, Mathematics, Natural Sciences or Humanities and Social Sciences, as well as technical and vocational training, through technical courses integrated with High School. These changes aim to guarantee a more meaningful education that is connected to the reality of young Brazilians, bringing teaching closer to the practical experience of students (Brasil, 2018).

Despite the advances brought about by this reform, mathematical learning still faces significant challenges, such as the low performance of students, the lack of continuing training for teachers and the disconnection between the content worked on in class and students' daily lives (Cysneiros, 2008). Faced with this situation, secondary education needs to overcome these obstacles, emphasizing that mathematics goes beyond numbers, letters, symbols, rules and techniques, and is an integral part of culture and human history.

Skovsmose (2000) emphasizes that the mathematics education must transcend techniques and algorithms, bringing it closer to everyday life and the social contexts in which students are inserted. This perspective reinforces the need for teaching practices that connect mathematical knowledge to students' experiences. The skills developed at this stage are fundamental to deepening mathematical literacy, promoting a more complete understanding of reality and enabling more precise and effective interventions.

According to the Common National Curriculum Base (BNCC).

> mathematical literacy is defined as: the skills and abilities to reason, represent, communicate and argue mathematically, in order to encourage the establishment of conjectures, the formulation and resolution of problems in a variety of contexts, using mathematical concepts, procedures, facts and tools. Literacy must also ensure that all students recognize that mathematical knowledge is fundamental to understanding and acting in the world and that they also perceive the intellectual playfulness of mathematics, as an aspect that favors the development of logical and critical reasoning, stimulates investigation and can also be pleasurable (enjoyment). (Brasil, 2018, p. 522)

In this scenario, the introduction of educational technologies, such as Computational Thinking, through interdisciplinary projects, is a promising alternative. This approach can not only reduce school dropout rates, but also make teaching more attractive, contextualized and aligned with the challenges of the 21st century.

Nóvoa (2017) emphasizes that schools that adopt innovative methodologies, adapted to students' needs, are able to create more attractive and meaningful environments, directly impacting the reduction of school dropouts and student engagement.

By integrating technological tools into the New High School curriculum, a more dynamic, contextualized and meaningful learning environment is created, favouring the development of transversal skills such as creativity, problem-solving and critical thinking, which are essential for preparing young people for the challenges of contemporary society.

According to Papert (1980), computer programming can transform learning into a more active and constructive process, allowing students to develop skills such as logical thinking, problem solving and creativity. This perspective reinforces the importance of using technologies as mediators in the educational process.

Along these lines, Moran, Masetto and Behrens (2018) point out that active methodologies, such as project-based learning and the use of educational technologies, play an essential role in promoting student protagonism. These practices allow students to actively participate in their education, contributing to contextualized and meaningful teaching, perfectly aligned with the demands of the New High School, strengthening the proposal to transform educational practice.

2 Computational Thinking in the BNCC

Like many scientific phrases and words, Computational Thinking is a reasonably well-known term in Brazil and around the world, although there are still doubts about its clear definition, its application and the disciplines that can benefit from it. Many believe that only areas such as Mathematics and Computer Science can benefit from this methodology, while others consider it a recent novelty, only widely disseminated in the last decade in academic circles.

Despite the nomenclature, Computational Thinking goes beyond computers, referring to a problem-solving mindset associated with mathematical thinking. Although it has often been linked to the mathematics education since the 1960s, it became widely recognized in basic education with the implementation of the BNCC, promulgated by the Ministry of Education (MEC) in 2017. According to the BNCC (2018).

> the area of mathematics in elementary school focuses on understanding concepts and procedures in its different fields and on developing computational thinking, with a view to solving and formulating problems in a variety of contexts. (Brasil, 2018, p. 471)

Computational Thinking is a set of skills and competencies that apply computational logic to solve problems in various areas of knowledge. It can be interpreted as the ability to analyze, model and solve complex problems, making it a valuable tool also in fields such as science, engineering, economics and society.

Wing (2006) defines Computational Thinking as a set of cognitive skills that allow us to solve complex problems, organize data and create computational models. She states that thinking should be considered a fundamental skill, comparable to reading, writing and arithmetic, and should be developed from childhood. Also according to Wing (2006), there is a growing realization that the 21st century job market requires computational skills. Furthermore, computational thinking can drive advances in various areas of knowledge.

Curzon *et al.* (2009) argue that Computational Thinking consists of skills organized into various cognitive and non-cognitive dimensions within an integrated framework. This approach fosters essential skills for solving complex problems, such as decomposition, abstraction, pattern recognition and algorithmic thinking. Brackmann (2017) defines Computational Thinking as a creative and critical skill, highlighting its role in creating innovative and efficient solutions.

For Barbosa and Maltempi (2020), Computational Thinking is one of the essential integrative skills to be developed during the mathematics education, which allows students to develop a more integrated view of the basic concepts of Mathematics. They also advocate the use of learning methods and strategies, such as problem solving, mathematical modeling and investigation, to create an environment conducive to the development of skills associated with both mathematical literacy and computational thinking.

Computational Thinking values new ways of thinking and exploring mathematical and computational knowledge, stimulating students' creativity and autonomy. The integration of computational and mathematical knowledge is essential for solving real problems and has great potential for social impact. This approach allows for the creation and understanding of sustainable electronic devices, applicable in different contexts, as well as enriching academic training, making it more dynamic and contextualized, while broadening the understanding of the role of technology in education and society (Azevedo & Araújo, 2023).

According to the BNCC (Brasil, 2018, p. 470), Computational Thinking in Primary School "focuses on developing an understanding of concepts and procedures in their different fields, with a view to solving problem situations". In secondary school, students use the same concepts they have acquired not only to solve problems, but also to design problems, describe and develop their own thinking, whether or not they use numerous resources other than mathematics. Brazil (2018) proposes that.

> students have been using technology such as calculators and spreadsheets since the early years of elementary school. This means that when they reach the final years,

> they can be encouraged to develop computational thinking by interpreting and drawing up flowcharts and algorithms. (Brasil, 2018, p. 518)

Also in the context of secondary school skills, computational thinking is explicitly addressed in some competences in the area of Mathematics. These include investigating and recording algorithms using flowcharts, when applicable, and applying the basic concepts of a programming language to implement algorithms described in mathematical or everyday language. According to Reis *et al.* (2021, p.43), “these two skills suggest focusing on the elaboration of algorithms and some possible ways of representing them: flowcharts, everyday language, mathematical language and programming language.”

Faced with these realities, the field of mathematics, its educators and its technologies have a responsibility to prepare students for processes of reflection and skill-building. This implies that students must use their previous potential to provoke these actions, directly affecting creative thinking, decision-making and analytical thinking in everyday situations.

3 Methodological Research Path

Although Computational Thinking is widely discussed and encouraged as an essential skill for the 21st century, studies verifying its real impact in the school environment are not easily found in the literature (Custumisu *et al.*, 2019). This gap in verifying the effectiveness of Computational Thinking and the development of skills, at all stages of education, contributes to teachers' misinformation.

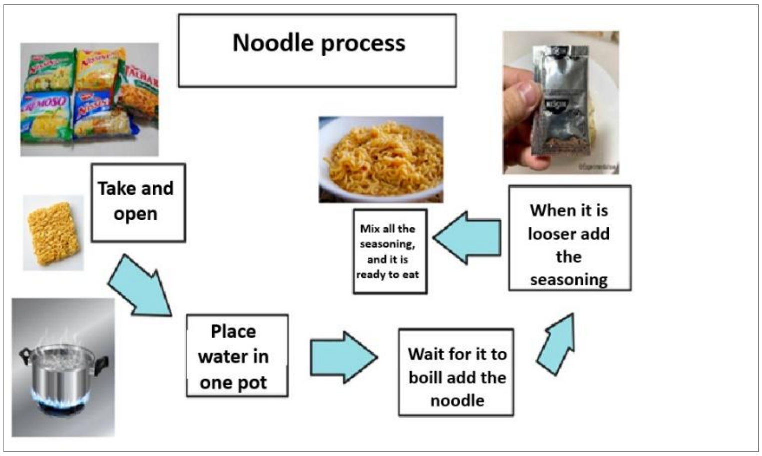
In view of these facts and with the aim of exploring and investigating the practical integration of computational thinking and its real effectiveness in the school curriculum context, a case study with a qualitative and interpretative approach was carried out. This study was structured to be implemented over the course of the 2023 school year, involving a class of 24 students from the 1st year of the New High School at a school located in the interior of the state of São Paulo.

In this context, the *Python* programming language can be used as a tool in the process of investigating, understanding and deepening mathematical concepts. With the aim of providing reflections, theoretical background and evaluating the impact of this approach in the school environment, it is argued that *Python* can be used to explore algorithms that identify numerical patterns, create computer visualizations that represent mathematical relationships and analyse data in depth. The choice of *Python* is justified by the simplicity and intuitiveness of its syntax, which facilitates the learning of logic and algorithms, as well as making it possible to create practical simulations that make abstract concepts concrete, promoting a broader and more interdisciplinary understanding (Marcondes, 2018; Grave, 2021).

The curriculum does not delve into computational thinking, but is limited to the integration of Mathematics and its Technologies and Computational Thinking in the Training Itinerary. In this context, the question arises: *how can programming languages be used to explore the concepts of Computational Thinking and Mathematics, going beyond the limits set by the curriculum?* Given this scenario, we propose an approach to mathematical concepts in the subject of Mathematics that can be modeled using simple programs developed by the students themselves.

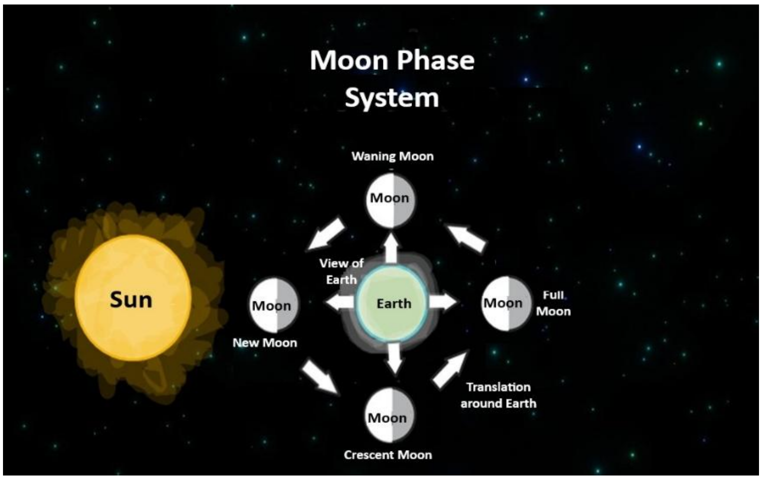
A possible starting point for addressing the relationship between theoretical content and its practical application involves introductory discussions about Computational Thinking, previously assimilated concepts and the basic functioning of computers. This approach can include exploring topics such as program structure, algorithms and systems, culminating in practical activities such as using collaborative tools like *Google Jamboard* to create non-computational systems based on everyday situations. In this dynamic, the algorithms for these systems can be developed and presented, promoting the correlation between the concepts discussed and their applications, strengthening the connection between theory and practice. For example, Figures 1 and 2 below show non-computational systems that reflect the students' own everyday experiences.

Figure 1: Instant noodle process system designed by the student A02.



Source: authors' archive (2025)

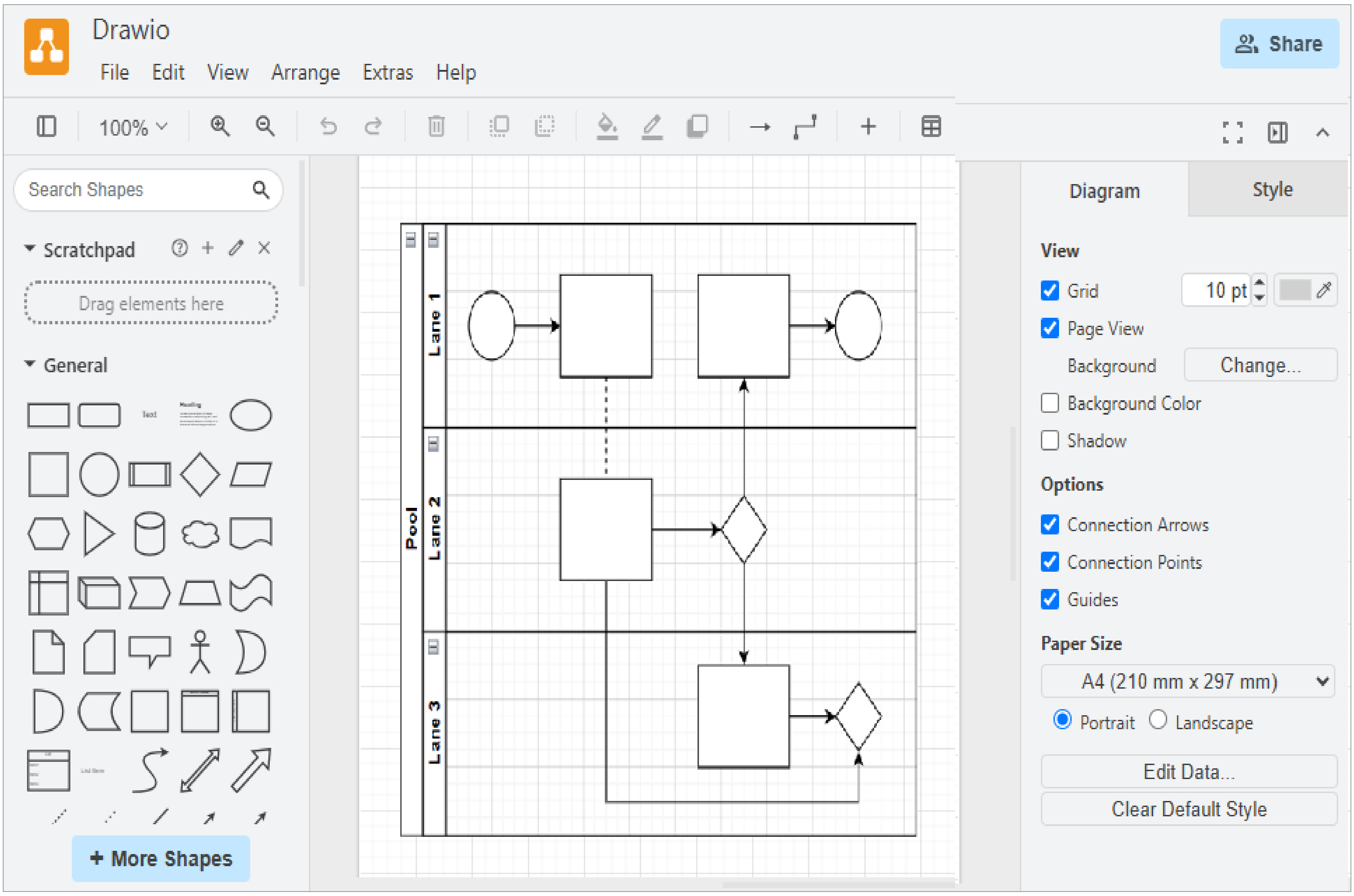
Figure 2: Moon phase system created by the student A04.



Source: authors' archive (2025)

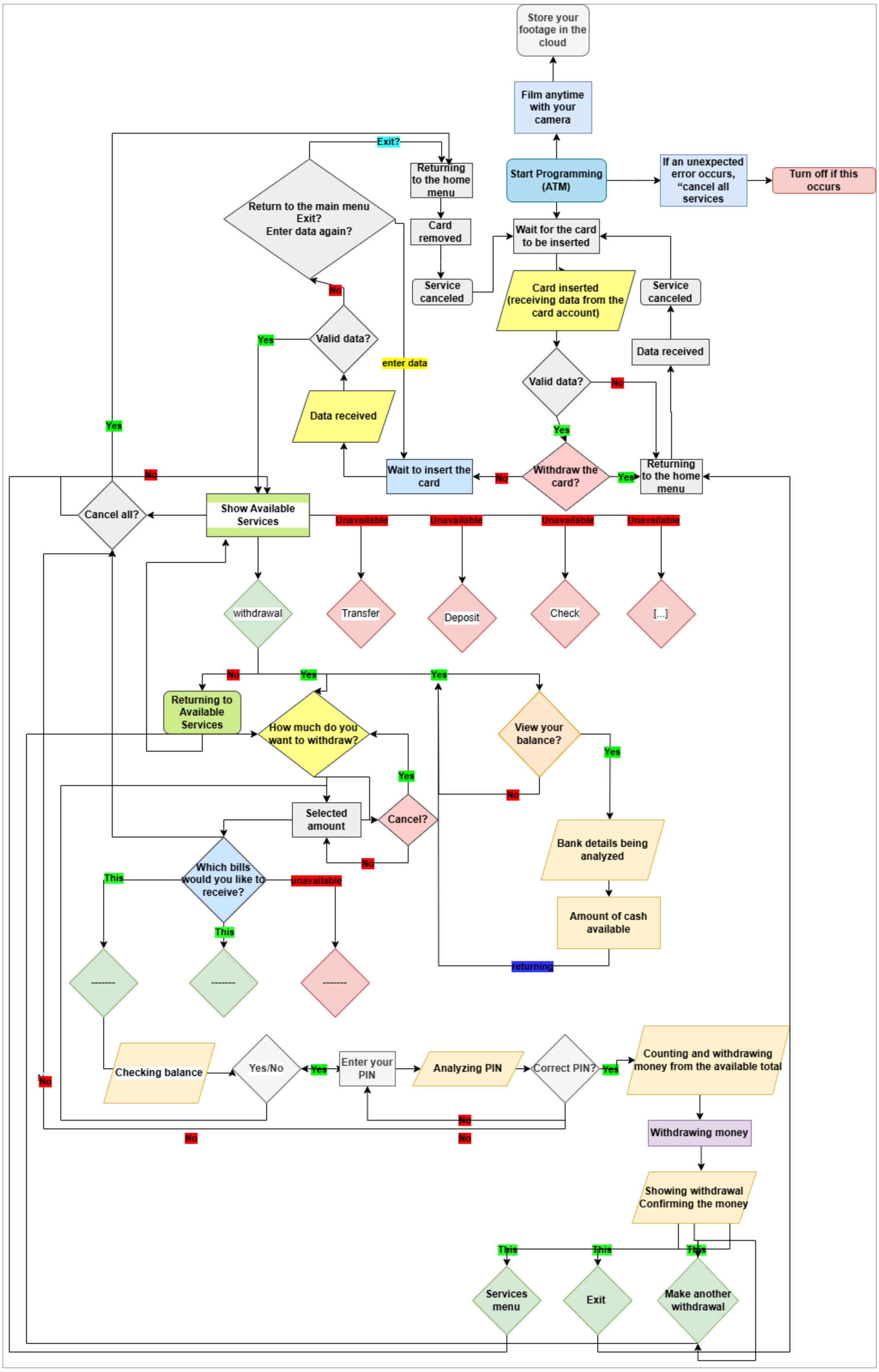
In order to prepare students for programming classes, an integrative approach can be used, connecting the study of Computational Thinking, logic and flowcharts to systems previously modeled in collaborative tools such as *Google Jamboard*. This practical and contextualized methodology can contribute to a deeper understanding of the concepts, facilitating the transition to the development of programming logic. In this context, tools such as *Draw.io*, an online platform with a simple and intuitive interface, can be introduced to create flowcharts and UML diagrams, such as sequences and mind maps, promoting the organization and structuring of logical thinking.

Figure 3: Draw.io tool start page



Source: *draw.io* website (2025).

Figure 4: Flowchart drawn up by student A04 for the activity 'thinking like an ATM'.



Source: authors' archive (2025)

Problem-based learning (PBL) can be used as a teaching methodology in which students, based on real problem situations, define objectives, search for information and discuss their findings, becoming the protagonists of their learning. In this approach, teachers act as facilitators, promoting critical and reflective thinking.

An example of applying this methodology might involve activities such as 'Thinking like an ATM', in which students create algorithms to represent the withdrawal process, using tools such as Draw.io to build flowcharts (Figure 4). In addition, the development of static HTML websites can be introduced as a practical and accessible way to start learning programming, favoring the construction of a dynamic and collaborative environment that encourages the spontaneous exploration of new topics.

Creating a static HTML website can be used as an introductory activity to teach basic programming concepts, covering the fundamental structure of a page and encouraging customization with text and images. This initial approach can start with simple examples, such as the classic 'Hello, World!' and be carried out in pairs, promoting not only technical learning, but also the development of soft skills, such as collaboration and communication (Figure 5).

Figure 5: HTML text markup - Hello, World!

```html
<!DOCTYPE html>
<html>
<head>

<style>
h1 {color:red;}

</style>
</head>

<body>
<h1>File header </h1>

<p>My first text: Hello, World!!!</p>


</body>
</html>
```

Source: authors' archive (2025)

Since the competences provided in the Computational Thinking subject have proved insufficient to meet the students' growing interest in programming at a later stage, the introduction of the *Python* language can be explored in an interdisciplinary way with the Mathematics and Physics subjects, using platforms such as *Replit* to facilitate online learning. The content can be structured gradually, covering topics such as Boolean algebra, data manipulation and flow controllers, with the inclusion of practical activities at the end of each topic to promote the consolidation of autonomous learning.

In order to promote the development of Computational Thinking, it is essential to take into account the students' prior knowledge of Mathematics and Physics, integrating it into the planning of activities. Programming can be used as a tool to reinforce this content, reviewing and consolidating previously established fundamental concepts. In *Python* classes, active methodologies can be used to put students at the center of the learning process, encouraging the exchange of information and experiences in productive pairs.

The content, in addition, can be presented in a structured way, combining practical activities with moments for discussion and experimentation. As an assessment strategy, interdisciplinary quizzes can be applied at the end of the activities to assess the practical application of math and physics concepts, integrating them with programming learning. Table 1 shows some guiding questions that can be used for formative assessment.

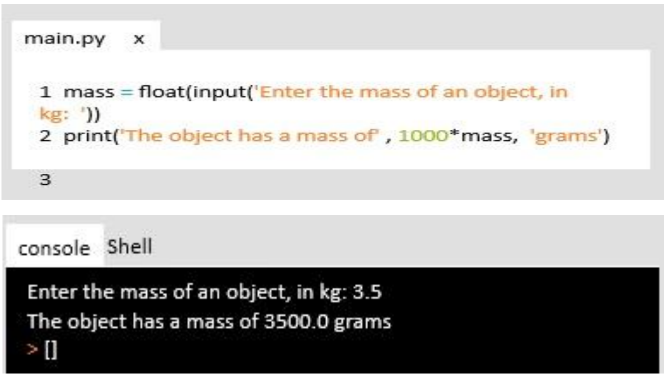
Table 1: Guiding questions for formative assessment.

1. Write a program that reads an integer and displays its successor and predecessor;
2. Write a program that reads two integers and shows the subtraction between them;
3. Write a program that reads three integers and shows the multiplication of the first two divided by the third;
4. Write a program that reads an integer and displays its double;
5. Write a program that reads any natural number and displays its full multiplication table;
6. Write a program that converts a mass given in kilograms to grams;
7. Write a program that converts a temperature given in Celsius to Fahrenheit;
8. Write a program that asks for the price of a good and the discount percentage and displays both the discount amount and the final price to be paid;
9. Write a program that asks for a user's year of birth and calculates how old they will be or will have been in 2025;
10. Write a program that asks for the distance (in kilometers) to be covered on a trip and the expected average speed and calculates the estimated time of the trip.

Source: self-authored (2025)

Formative assessment can be used as a strategy for students to present their results in teams, making it possible to integrate knowledge of physics and programming. Questions involving physics concepts can be particularly relevant, as they allow students not only to develop programming skills, but also to apply physics concepts, correctly associating and verifying the real values and numbers used in the programs. This approach encourages the analysis and checking of data in real-world contexts, promoting interdisciplinary and meaningful learning.

Figure 6: Screenshot of student A01 solving Question 6.



Source: authors' archive (2025)

Over time, programming, which was previously not part of students' daily lives, can gain a significant place in their learning. Although challenges may arise, it is possible to observe the persistence of the students as they dedicate themselves to solving problems and correcting errors in the codes they develop.

Figure 7: **(**Translated reproduction of**)** Resolution of question 10 by student A03, initially written on paper before transcribing and executing the commands in programs. Original in portugueses.

```
10) distance=float(input("What is the distance to
your dstination, in kilometers?")

speed = (input("What is the expected average speed
for the trip, in kilometers per hour?")

print("The trip willtake approximately, distance/speed,
'hours')

END
```

Source: authors' archive (2025)

The Fibonacci sequence, in particular, arouses great curiosity in classes, especially because of the appearance of the number Phi (φ), also known as the golden ratio, and its various applications in everyday life. An example of an activity that can be carried out is the creation of a calculator for the Fibonacci sequence, which accurately calculates the requested term, starting with the digit zero. Student A04, for example, managed to associate the programming language with a topic that required mathematical skills: progressions and sequences (Figures 8 and 9).

Figure 8: Extra work by student A04 - Fibonacci calculator.

```
▶ Run
main.py ×  +
main.py > N                                                        Format
N = int(input(" Enter an index of the Fibonacci series, starting of zero "))

if N== 0:
  print("  The first term of serie is 1   ")
elif N ==1:
  print ("  The second term of serie is 1  ")
else:
  penultimateterm  = 1
  lastterm = 1
  term  = 0
  i=2
  while i <=N:
    term = penultimateterm + lastterm
    penultimateterm = lastterm
    lastterm = term
    i = i + 1

print ("The",N, "º term of serie is", term )
```

Source: authors' archive (2025)

Figure 9: Console showing the execution of the program created by student A04.

```
>_ Console    Shell    Deployments    +

Run                                   Ask AI   7s on 09:40:16, 05/28 ✓

Enter an index of the Fibonacci serie, starting at zero 023
The 23 ° term of the serie is 46368

Run                                   Ask AI   4s on 09:41:08, 05/28 ✓

Enter an index of the Fibonacci serie, starting at zero 054
The 54 ° term of the serie is 139583862445
```

Source: authors' archive (2025)

Figure 10: Extra work by student A06 on measurement conversions.

```
print("Conversion from kilograms")
q = float(input(" Enter the quantity in kilograms :"))
ton = q / 1000
gra = q * 1000
tond = q / 1016
tonc = q / 907.2
sto = q / 6.35
lib = q * 2.205
onç = q * 35.274

print(" The value in tons :", ton)
print("The value in grams", gra)
print(" The value in short tons :", tond)
print(" The value in Short tons :", tonc)
print("The value in Stone:", sto)
print("The value in Pound:", lib)
print(" The value in Ounce :", onç)

print("Conversion from centimeteres :")
cms = float(input(" Enter the quantity em centimeteres :"))
mm = cms * 10
dm = cms / 10
m = cms / 100
dam = cms / 1000
hm = cms / 10000
km = cms / 100000

print("The value in milimeteres:", mm)
print("The value in decimeters:", dm)
print(" The value in meters :", m)
print(" The value in dectameteres :", dam)
print(" The value in hectometers :", hm)
print(" The value in kilometers :", km)
```

Source: authors' archive (2025)

It can be seen in Figure 10 that student A06, in addition to performing the conversions requested in the assessment (Question 7 of Block 1) very accurately, also went beyond expectations by spontaneously presenting other conversions involving different units of measurement, such as mass and length. Her initiative not only surprised the researchers, but also revealed an in-depth mastery of the content, as well as her proactive, curious and independent attitude to the learning process.

In view of the above, the next section presents a qualitative and interpretative analysis of the research, focusing on the practical integration of computational thinking and the evaluation of its effectiveness in the context of the school curriculum.

4 Discussions and Data Analysis

Programming classes can be concluded with the application of an assessment rubric that covers all the content taught over the course of the school term. As part of the reflective process, interactive tools such as *Jamboard* can be used for students to share their perceptions of the lessons, highlighting the most appreciated aspects, the lessons learned and the relevance of the content in their lives.

Rubrics, in this context, have the main objective of evaluating and monitoring student performance using well-defined scales and criteria, offering a structured and coherent analysis. In addition, these tools are versatile and can be adapted to different educational scenarios. In this sense, Fernandes (2021) points out that.

> in the context of formative assessment, assessment for learning, i.e. to provide high quality feedback, or in the context of summative assessment, assessment of learning, so that, at a given point in time, we can take stock or take stock of what the students know and are able to do. (Fernandes, 2021, p. 4)

Rubrics can therefore be used as a pedagogical tool to strengthen the teaching-learning process. Through them, it is possible to collect qualitative information about students' performance, their individual perceptions and the relevance of an educational project in their learning trajectories. This approach enables an in-depth analysis of the students' reflections, offering a clearer view of the impact of the activities carried out.

In addition, the use of rubrics can facilitate the feedback process, allowing for continuous adjustments and improvements in the educational process. They can be developed based on specific questions, with the aim of analyzing students' perception and performance in projects related to Computational Thinking and the *Python* Language.

Table 2 below sets out the evaluation rubric adopted in this study in a detailed and structured way, presenting the questions and objectives that guided the participants' evaluation and reflection process throughout this study. This enables an in-depth analysis of the learning and experiences gained during the research.

Table 2: Evaluation rubric (the questions and their objectives).

Questions	Objectives
How would you rate the knowledge you have acquired during these activities?	Promote reflection on the knowledge acquired and the applicability of the content covered throughout the activities.
What aspects of the activities stood out for you in terms of interest and relevance?	Identify the contents or activities that aroused the most interest, justifying the reasons for this relevance.
What were the main challenges encountered during the development of the activities?	Analyze the main challenges faced during the process, providing a detailed description of the difficulties encountered.

What were your initial expectations of the activities and how would you assess the dynamics of the classes?	Evaluate the initial expectations regarding the activities and comment on the dynamics, methodology and quality of the classes offered.

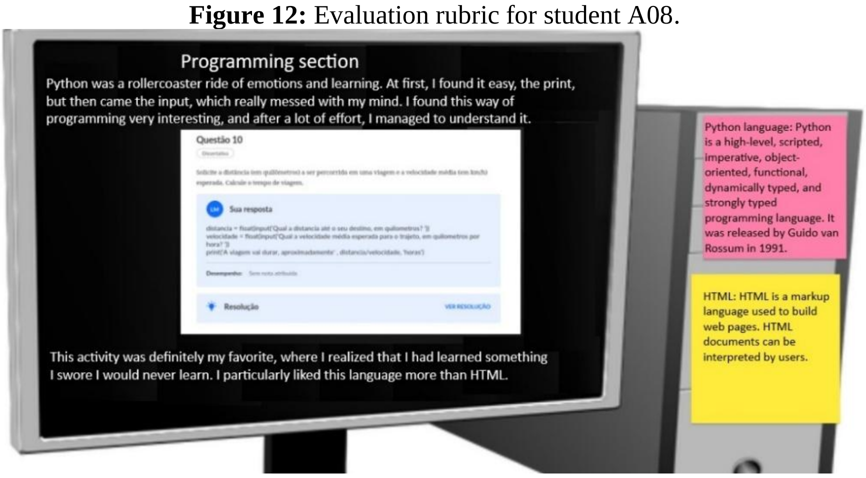
Source: self-authored (2025)

The structure adopted made it possible to collect relevant qualitative data, allowing for a more in-depth analysis of the students' experiences, challenges and achievements throughout the project and activities. It was observed that the students provided sincere reflections on their participation, highlighting the relevance of the project to both their learning of mathematics and physics. From these reflections, positive aspects were identified, such as increased interest and appreciation of the content covered, as well as the challenges faced throughout the process. Some students mentioned specific difficulties and expressed preferences regarding the methodologies and dynamics of the classes, highlighting the diversity of perceptions and learning styles present in the class. This data provides valuable information for understanding the impact of the project and for continually improving teaching practices.

Figure 11: Student evaluation rubric A07.

Source: authors' archive (2025)

Figure 12: Evaluation rubric for student A08.



Source: authors' archive (2025)

Some students faced initial difficulties with programming, particularly due to the language barrier imposed by the use of English in the *Python* language and a lack of familiarity with the technology area. Despite these limitations, which represented challenges, the class managed to make significant progress.

Most of the students showed resilience, being able to assimilate the concepts and delve into the proposed content effectively. This process highlights not only the students' commitment, but also the importance of inclusive and adaptive teaching practices that encourage learning, even in the face of individual challenges[1]:

Student A10 [avaliative rubric]: *"Python language was a rollercoaster of emotions and learning, at first I found it easy, the print, then came the input that messed with my mind. I found this way of programming very interesting, and after a lot of effort I managed to understand it."*

One of the students, identified as A11, when answering the rubric, reflected on the knowledge acquired in the final phase of the project, highlighting not only the content learned, but also its practical applications. The student noted that he had mastered basic *Python* language commands, such as input and print, which make it possible to enter and display information on the computer. In addition, the student showed an understanding of the use of mathematical operators and the creation of tools such as simple calculators, scientific calculators and unit converters:

Student A11: [avaliative rubric]: *"Basically, I learned a bit about the python programming system, then I learned the basic commands, such as input, which is a command that allows you to give information to the computer [...] after that, I also learned about the print command, which is used to "print" what was mentioned before [...] I also learned that when you mention a number, you just need to type it in brackets.] I also learned that when you mention a number you only need to type it in brackets, but if you mention a sentence you need to use brackets and quotation marks, and that when you mention an integer, you should use the int command [...] I also learned that you can perform mathematical operations inside Python, i.e. you can make a calculator for many things inside this type of programming, such as a simple calculator, a scientific calculator, a calculator for degrees Fahrenheit and Kelvin, interest, multiplication tables and many more."*

It is clear from the answers that the students were not only interested, but also recognized the importance of interdisciplinarity between mathematics and physics. They understood how the concepts of these disciplines interrelate and complement each other, promoting a more in-depth and integrated view of the topics covered:

Student A12 [avaliative rubric]: *"Python can also be useful for other subjects such as physics [...] we can calculate the time needed for a journey by knowing the average speed and distance."*

The development of Computational Thinking skills was significantly expanded over the course of the project, especially since many students were already familiar with programming and used it in their daily lives. This previous experience made it easier to delve deeper into the content, allowing students to apply and connect new concepts with knowledge they had already acquired:

[1] The answers obtained were transcribed verbatim, without undergoing grammatical correction criteria, in order to guarantee a reliable analysis of the phenomena investigated in this research.

Student A13 [avaliative rubric]: *"I've been programming since 2021 and I have all the basic and introductory knowledge of programming, mainly focused on programming logic, so I can easily learn any language. [...] I had never studied HTML [...] and Python, which we covered recently. [...] The teacher introduced basic programming concepts, such as variables, like string, int and float. She did a great job of teaching programming through practical work, such as creating a calculator, which teaches arithmetic operations and int and float variables very well, and also fitting in physics, it was very good. [...] In this bimester I created a fully functional calculator that performs all the arithmetic operations."*

In addition, the Computational Thinking classes enabled in-depth reflection on the profile of professionals working in this area, as well as the real functions they perform, challenging and demystifying paradigms and misperceptions that the students had about the profession:

Student A14 [avaliative rubric]: *"The programmer is not simply the bespectacled person who sits behind a computer, in fact, this reading of the professional is a stereotype. [...] The programming language studied during the chapter was Python, which is simple to learn and can be used to solve complex problems. I particularly liked this type of programming."*

After applying the evaluation rubric, an exchange of reflections between the students was promoted, allowing each one to share the learning acquired during the project, as well as the difficulties faced throughout the learning process. The researchers led the final reflection, highlighting the results obtained by the class and showing both individual and collective progress.

This closing followed the principles proposed by Polya (1995), who emphasizes the importance of understanding the problem, formulating an action plan, carrying it out and, finally, carrying out a retrospective analysis of the process. This moment was essential for consolidating the skills and abilities developed, enabling students to reflect on their progress and the practical application of the knowledge acquired.

5 Final Considerations

The aim of this study was to investigate the impact of integrating Computational Thinking into the mathematics education in the New High School, through a subject in the Formative Itinerary. A case study was carried out in one class, with the aim of analyzing whether the use of playful and technological elements, such as programming languages, can contribute to a more effective learning environment, promoting the development of students' logical reasoning.

Integrating computational thinking into high school math classes has proven to be an effective strategy for increasing engagement and improving student performance. This work showed that the introduction of programming concepts and computational logic not only favored the development of logical reasoning, but also improved students' problem-solving skills and capacity for abstraction, providing more meaningful learning in line with contemporary demands.

The classes were dynamic and challenging, promoting the active involvement of the students in each stage of the activities. Using technological tools and exploring programming environments for writing in *Python*, the students took part in a practical and interactive approach, which favored the application of theoretical knowledge in concrete contexts, making learning more relevant and efficient. Most of the students pointed out that the method not only facilitated their understanding of mathematical concepts, but also made learning more engaging and meaningful, connecting it in a way that was relevant to their everyday lives and professional prospects. The evaluation rubrics developed by the students themselves revealed great appreciation for the computational thinking project and the use of the *Python* programming language.

The results of the assessments and evaluation rubrics indicated that the students not only understood the mathematical concepts presented better, but also showed greater interest and motivation in the proposed activities, especially due to the interdisciplinary integration with Physics. The interdisciplinarity between mathematics and physics proved to be highly productive, allowing students to explore physical concepts in a more intuitive and practical way through computational thinking. A significant example was the programming of simulations for measurement transformations, which facilitated the understanding of topics such as kinematics and dynamics. This integrated approach not only consolidated knowledge in both disciplines, but also highlighted the applicability of mathematics in solving real problems in the context of physics.

Furthermore, collaboration between math and physics teachers played a key role in the success of this project. Working together, the educators developed more coherent and interconnected projects, demonstrating to the students the relevance and interdependence of scientific knowledge. This synergy also promoted a rich exchange of teaching methodologies and strategies, enriching teaching practice and expanding the repertoire of resources available for teaching, with significant benefits for student learning.

The process, however, was not without its challenges. The applying teacher-researcher faced considerable difficulties, especially related to a lack of familiarity with technological tools and the need for ongoing training to master new teaching methodologies. Integrating Computational Thinking into the curriculum required a significant effort to adapt teaching materials and create activities that were both challenging and accessible to all students, as well as being aligned with the skills developed in the Mathematics and Physics subjects.

Thus, it is essential to invest continuously in training educators so that they can take ownership of the technologies and methodologies needed to effectively implement Computational Thinking. Likewise, it is essential to guarantee adequate infrastructure in schools, with access to computers and programming software, so that all students can fully participate in the proposed activities. In this sense, the research needs to continue in order to verify the weaknesses in teacher training, as well as in school infrastructure, so that further training can be developed more effectively. This is why it is essential to carry out qualitative-quantitative research to gather information to be categorized, analyzed and interpreted.

This study indicates that Computational Thinking is a promising approach to teaching mathematics in secondary schools, in line with both the guidelines of the National Common Curriculum Base (BNCC) and the demands of the 21st century. For future research, it is recommended to investigate the approaches and impacts of Computational Thinking in different educational scenarios so that new proposals can be developed within the mathematics education and other teaching areas. The experience reported suggests that the integration of technological and playful elements can create a more enriching and motivating learning environment, capable of developing essential skills in students. In turn, they realized that mathematics is present in content that was previously considered challenging, but which can be learned effectively through interdisciplinarity, by solving problems in a detailed and contextualized way.

Acknowledgements

This paper was carried out with the support of the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES) - Funding Code 001.

References

Azevedo, G. T. & Araújo, U. F. (2023). Desenvolvimento científico–robótico no âmbito da formação em matemática: pensamento computacional e relevância social. *Revista Internacional de Pesquisa em Educação Matemática (RIPEM), 14*(1), 1–17. https://doi.org/10.37001/ripem.v14i1.3706

Barbosa, L. L. S. & Maltempi, M. V. (2020). Matemática, Pensamento Computacional e BNCC: desafios e potencialidades dos projetos de ensino e das tecnologias na formação inicial de professores. *Revista Brasileira de Ensino de Ciências e Matemática (RBECM), 3*(3), 748–776. https://doi.org/10.5335/rbecm.v3i3.11841

Brackmann, C. (2017). *Desenvolvimento do Pensamento Computacional através de atividades desplugadas na Educação Básica* [Dissertação de Doutorado, Universidade Federal do Rio Grande do Sul].

Brasil. Ministério da Educação. (2018). Base Nacional Comum Curricular: Resolução CNE/CP nº 4, de 17 de dezembro de 2018.

Curson, P., Black, J., Meagher, L. R. & McOwan, P. W. (2009). cs4fn.org: Enthusing students about computer science. *Proceedings of Informatics Education Europe IV*, 73–80.

Cutumisu, M., Adams, C. & Lu, C. (2019). A scoping review of empirical research on recent computational thinking assessments. *Journal of Science Education and Technology, 28*(6), 651–676. https://doi.org/10.1007/s10956-019-09799-3

Cysneiros, P. G. (2008). PAPERT, Seymour. A máquina das crianças: repensando a escola na era da informática. *Revista Entreideias: Educação, Cultura e Sociedade, 12*(12), 227–231. https://doi.org/10.9771/2317-1219rf.v12i12.2971

Fernandes, D. (2021). *Rubricas de avaliação*. Ministério da Educação/Direção-Geral da Educação.

Grave, L. A. S. (2021). *O pensamento computacional na prática: uma experiência usando Python em aulas de Matemática básica* [Master's thesis, Universidade Federal de Santa Maria].

Marcondes, G. A. B. (2018). *Matemática com Python–um guia prático*.

Masola, W. & Allevato, N. (2019). Dificuldades de aprendizagem matemática: algumas reflexões. *Educação Matemática Debate, 3*(7), 52–67. https://doi.org/10.24116/emd.v3n7a03

Moran, J., Masetto, M. T. & Behrens, M. A. (2018). *Novas tecnologias e mediação pedagógica: O papel das metodologias ativas na educação contemporânea*. Editora UNESP.

Nóvoa, A. (2017). Firmar a posição como professor, afirmar a profissão docente. *Cadernos de Pesquisa, 47*(166), 1106–1133. https://doi.org/10.1590/198053144843

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

Polya, G. (1995). *A arte de resolver problemas: um novo aspecto do método matemático*. Interciência.

Reis, S. R., Barichello, L. & Mathias, C. V. (2021). Novos conteúdos e novas habilidades para a área de Matemática e suas Tecnologias. *Revista Internacional de Pesquisa em Educação Matemática (RIPEM)*, *11*(1), 37–58. https://doi.org/10.37001/ripem.v11i1.2539

Skovsmose, O. (2000). Cenários para Investigação. *Bolema*, *13*(14), 66–91.

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