ANNALS OF "DUNAREA DE JOS" UNIVERSITY OF GALATI
MATHEMATICS, PHYSICS, THEORETICAL MECHANICS
FASCICLE II, YEAR XVII (XLVIII) 2025, No. 2
DOI: https://doi.org/10.35219/ann-ugal-math-phys-mec.2025.2.13

Developing computational thinking through playful activities in lower secondary education

Daniela - Mariana Crețu[1], Alina Gabriela Țepeș [1], Georgeta Neicu[1], Simona Elena Humelnicu Christofi[1]

[1] Gymnasium School no. 28, 31 Rosiori St., 800044 Galati, Romania
[*] Corresponding author: daniela.cretu@yahoo.com

Abstract
This paper presents a theoretical analysis of methods for developing computational thinking in middle school students through the use of playful activities integrated into mathematics teaching. Computational thinking represents a set of essential cognitive skills for problem-solving, with transversal applicability across STEM fields. The paper highlights how playful activities can support the development of computational thinking in middle school students (grades 5 and 6), with a focus on mathematics instruction. Computational thinking is a key 21st-century skill that helps students approach complex problems and develop logical and algorithmic strategies.
The study presents concrete examples of activities carried out during Code Week, such as "Pete the Cat Calculates Gauss's Sum" and "Draw Geometric Shapes with Code – Mathematics Comes to Life in Scratch!", demonstrating how both digital and non-digital games can be integrated into mathematics lessons to stimulate understanding of concepts, as well as students' motivation and creativity. The study emphasizes the importance of playful approaches in fostering curiosity, intrinsic motivation, and creativity, providing methodological guidance for designing educational activities centered on active and collaborative learning. The conclusions underline the necessity of intentionally integrating educational games into the process of developing computational skills in the context of mathematics teaching at the middle school level.

Keywords: computational thinking, playful activities, mathematics education, digital competences, lower secondary education.

1. INTRODUCTION

Schools play a crucial role in preparing students for future challenges. Contemporary, deeply digitalized societies demand the development of higher-order thinking skills that enable individuals to adapt to new forms of interaction, learning, and problem-solving. Among these, computational thinking occupies a central role, being defined as an approach to complex situations through concepts and strategies inspired by computer science, such as problem decomposition, pattern recognition, abstraction, and algorithmic thinking.

In lower secondary education, mathematics provides an ideal framework for developing these abilities, as it naturally involves logical reasoning, algorithmic structures, and generalization processes. Integrating playful activities into mathematics lessons not only increases student engagement but also facilitates cognitive transfer between logical and computational thinking.

The purpose of this paper is to analyze, from a theoretical perspective, how playful activities can be used as effective tools for cultivating computational thinking among lower secondary school students.

2. THEORETICAL FRAMEWORK

The concept of computational thinking was introduced by Jeannette Wing (2006) as a key competence of the 21st century, comparable to reading, writing, and arithmetic. According to her, computational thinking involves “formulating problems and their solutions in a way that a human or a computer can effectively carry out.”

Recent research (Grover & Pea, 2013; Bocconi et al., 2022) shows that developing computational thinking should begin as early as lower secondary school, through interdisciplinary activities that combine logic, problem-solving, visual representations, and programming. In this context, educational games become a natural learning tool.

Therefore, within the context of mathematics education, playful activities create a cognitive experimentation environment that stimulates key processes of computational thinking. According to constructivist theory (Piaget, Vygotsky), playful activities support active knowledge construction, develop intrinsic motivation, and create authentic contexts for collaborative learning.

Therefore, in the context of mathematics education, playful activities create a cognitive experimentation environment that stimulates key processes of computational thinking, such as:

- analyzing and decomposing problem situations;
- formulating algorithms for problem-solving;
- evaluating and optimizing strategies;
- formally communicating solutions.

3. PLAYFUL ACTIVITIES FOR DEVELOPING COMPUTATIONAL THINKING

In the specialized literature, educational games are generally classified as logical-mathematical, digital, and interdisciplinary.

At the lower secondary level, they facilitate the transition from procedural learning to strategic thinking.

Playful activities can be grouped into the following three main categories:

- Logical-mathematical games – Sudoku, numerical puzzles, speed-based challenges;
- Digital games – Code.org, Lightbot, Scratch, Tynker;
- Interdisciplinary games – combine mathematics, computer science, and visual arts.

These games support the transition from procedural learning to strategic and algorithmic thinking while also stimulating creativity and collaboration among students.

Relevant examples include:

- Pete the Cat Calculates Gauss’s Sum (Fig.1) – students explore the sum of consecutive numbers through an interactive digital activity. The activity allows them to discover Gauss’s rule and practice abstract thinking, pattern recognition, and simple algorithmic strategies.

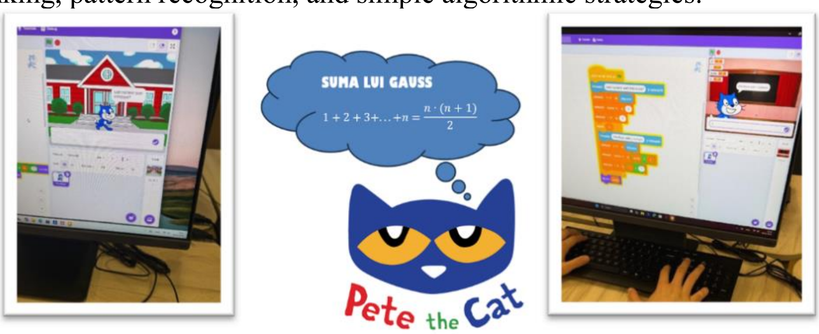

Fig. 1. Photos from the activity "Pete the Cat Calculates Gauss’s Sum"

• Draw Geometric Shapes with Code – Mathematics Comes to Life in Scratch! – students' program geometric shapes, learn concepts of related angles, symmetry, and recursion. This activity strengthens the link between geometry and algorithms while developing sequential and logical thinking (Fig.2).

Fig. 2. PHOTOS FROM THE ACTIVITY "DRAW GEOMETRIC SHAPES WITH CODE – MATHEMATICS COMES TO LIFE IN SCRATCH! "

• The "Secret Algorithm" game – students receive a sequence of steps (described in natural language) and must identify the general rule that governs the process. This develops abstraction and pattern-recognition skills.

• "Fractal Race" – a competition between teams building recursive geometric models, training iterative reasoning and understanding of recursion.

• "Numeric Code" – a simple encryption/decryption activity based on numeric correspondences, allowing students to practice binary logic and functional relationships.

• Digital educational games (e.g., Code.org, Lightbot, Tynker) – offer interactive visual environments for exploring with algorithmic concepts in an accessible and engaging way.

Integrating these activities into mathematics lessons supports the formation of algorithmic cognitive strategies and the understanding of the relationship between mathematical modeling and logical programming (Table 1).

Learning Objective	*Playful Activity*	*Implementation Suggestions*
Develop algorithmic thinking	**Pete the Cat Calculates Gauss's Sum**	Students work individually or in pairs to discover the sum of consecutive numbers and explain the algorithm in their own words.
Learn geometric concepts through coding	**Draw Geometric Shapes with Code – Mathematics Comes to Life in Scratch!**	50-minute session: 15 min explanation, 30 min coding and shape creation, 5 min group presentation.
Pattern recognition and problem analysis	**The Secret Algorithm**	In small groups, students receive a sequence of steps and formulate the governing rule; final discussion on strategies.
Iterative reasoning and recursion	**Fractal Race**	Team competition: construct the correct fractal quickly; discussion of steps and strategies.
Numeric logic and functional relationships	**Numeric Code**	Students encrypt simple messages, exchange with peers, and decrypt; discussion of connections to mathematical operations.

Table 1. Examples of Activities

Methodological recommendations for teaching mathematics suggest integrating playful activities with lesson objectives so that students learn through play and understand mathematical concepts more effectively. Alternating theoretical explanations with practical exercises maintains student attention and allows immediate application of knowledge.

For instance, in the lesson ‘Perfect Squares, Perfect Cubes,’ a worksheet was used featuring square and cube figures designed for overlaying (Fig. 3). Students first identified perfect squares from 1 to 10 to reinforce visual recognition of the concept. They then calculated perfect squares from 11 to 20 and perfect cubes from 1 to 14, applying calculation algorithms and mathematical reasoning. The combination of practical, visual, and abstract exercises created a strong link between theory and application, stimulating logical thinking, attention to detail, and autonomy in problem-solving. Evaluation emphasized reasoning processes, strategic thinking, and problem-solving approaches, encouraging critical thinking and algorithmic reasoning.

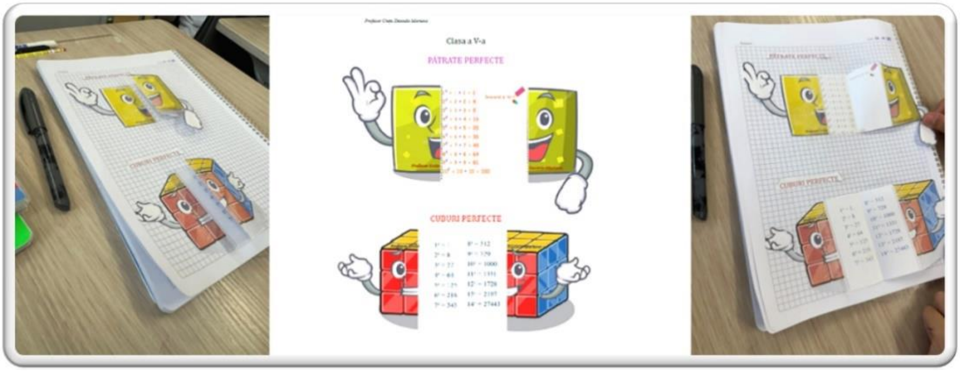

Fig. 3. PHOTOS FROM THE ACTIVITY "DRAW GEOMETRIC SHAPES WITH CODE – MATHEMATICS COMES TO LIFE IN SCRATCH! "

4. DISCUSSION AND DIDACTIC IMPLICATIONS

The theoretical analysis conducted shows that playful approaches can generate deep learning by simultaneously engaging motivation, analytical thinking, and collaboration among students. Playful activities produce a paradigm shift in mathematics teaching, from knowledge transmission toward discovery through exploration and modeling.

Effective implementation of such strategies requires teacher training in computational thinking and the integration of digital content into the curriculum. Moreover, learning assessment should target competences such as algorithmic reasoning, critical thinking, and mathematical creativity, not merely procedural performance.

Consistent implementation requires teacher training in computational thinking and integration of digital content into the curriculum. Assessment should focus not only on calculation performance but also on algorithmic reasoning, critical thinking, and mathematical creativity. Teachers play a key role in facilitating activities, creating a safe exploration environment, and connecting digital tasks to mathematical objectives. Code Week activities demonstrate that students in grades 5 and 6 can understand complex mathematical concepts and apply them digitally, with high engagement and satisfaction.

5. CONCLUSIONS

Computational thinking represents a core component of digital and mathematical literacy in 21st-century education.

Playful activities provide a valuable pedagogical framework for stimulating this competence, transforming learning into an exploratory and motivational experience.

Observations during activities indicate that students were more motivated and focused when lessons were structured as games, both digital and non-digital. Activities such as “Pete the Cat Calculates Gauss’s Sum” captured students’ attention through interactive stories and subtle competition, stimulating the desire to solve problems and understand numeric rules. Students actively engaged, asking questions like “What if we do it this way?” or “How can we generalize?”, demonstrating that play fostered critical thinking and curiosity.

Activities like The Secret Algorithm and Fractal Race promoted teamwork, idea exchange, and argumentation of reasoning. Students learned to communicate strategies, listen to each other, and negotiate solutions, strengthening social and collaborative skills. This group involvement was more intense than in traditional lessons, with students motivated by competition and the desire to complete challenges correctly.

Students showed initiative in discovering algorithms and mathematical rules. For example, in Draw Geometric Shapes with Code – Mathematics Comes to Life in Scratch!, some students created additional geometric patterns beyond the assigned task, experimenting with angles and recursion. This demonstrates that play and visual elements stimulate independent exploration and a desire to learn through experimentation.

Students’ reactions indicate that playful activities facilitate the transition from intuitive to logical and algorithmic thinking. They began structuring problem-solving steps, testing hypotheses, and formulating clear rules, reflecting the transfer of mathematical knowledge to digital and real contexts.

Game-based activities reduced performance pressure and fear of mistakes. Students felt freer to experiment and try different strategies, which increased confidence in their problem-solving abilities.

Students expressed that playful lessons were “more interesting” and “easier to understand” compared to traditional methods. Games made abstract concepts tangible and gave them practical applications, increasing curiosity and interest in mathematics.

Active involvement and positive experiences in these activities may have long-term effects on students’ perceptions of mathematics and computer science. They are more likely to apply algorithmic strategies in problem-solving, explore new concepts, and adopt critical thinking in various contexts.

This paper demonstrates, from a theoretical standpoint, that integrating educational games into mathematics teaching at the lower secondary level contributes to the development of complex cognitive competences, increases interest in problem-solving, and strengthens the connection between mathematics and computer science.

References

1. Bocconi, S., Chioccariello, A., Earp, J. Developing Computational Thinking in Compulsory Education – European Framework, European Schoolnet, 2022.
2. Grover, S., Pea, R. Computational Thinking in K–12: A Review of the State of the Field, Educational Researcher, 42(1) (2013) 38–43.
3. Papert, S. (1980). Mindstorms: Children, Computers, and Powerful Ideas. Basic Books.
4. Vygotsky, L. Mind in Society: The Development of Higher Psychological Processes. Harvard University Press, 1978.
5. Wing, J. Computational Thinking, Communications of the ACM 49(3) (2006) 33–35.