

The ComeThinkAgain Project: Playful Computational Thinking in Primary Teacher Education Through Game Mechanics

Bernadette Spieler and Melanie Kieber

Centre for Media Education and Informatics, Zurich University of Teacher Education, Switzerland

bernadette.spieler@phzh.ch

melanie.kieber@phzh.ch

Abstract: The Erasmus+ project "ComeThinkAgain" addresses the growing need to equip future generations with interdisciplinary skills by developing a Certification-based Education Training System (CETS) that integrates Computational Thinking (CT), Entrepreneurship Education (EE), and Green Skills (GS). For CT, a significant challenge in primary education is the training of teachers, who frequently lack a formal computer science background, to confidently teach foundational CT concepts. To address this issue, a 1 ECTS micro-module for CT foundations was developed specifically for primary school teachers, utilising a game-based learning (GBL) and playful approach. This paper presents the pedagogical design and initial pilot results of the course. The course followed a constructionist, hands-on modelling approach aimed at translating abstract computational concepts into tangible and relatable learning experiences. Participants engaged with theoretical reading tasks, analysed familiar games and everyday situations, and explored practical workshop stations involving unplugged, gamified, and tangible activities. These activities were designed to make CT concepts such as decomposition, abstraction, pattern recognition, algorithmic thinking, sequencing, and debugging visible without placing programming at the centre of the learning process. In the final transfer phase, teachers applied these insights by designing their own cross-curricular CT lesson activities. The pilot course was conducted with a cohort of five primary school teachers. The present study collected qualitative feedback from focus group and individual interviews as well as teacher-designed lesson artefacts. The aim was to explore how participants experienced the module, how they conceptualised CT, and whether they were able to transfer CT concepts into their own teaching contexts. The findings suggest that familiar game environments and tactile unplugged activities helped participants reframe CT as a broader problem-solving approach. Participants reported increased awareness of CT in everyday teaching practices and demonstrated initial transfer through the design of cross-curricular lesson activities.

Keywords: Computational thinking, Game-based learning, Teacher education, Game mechanics, ComeThinkAgain

1. Introduction

Computer science education can no longer be viewed merely as the teaching of programming skills or basic, application-oriented digital competencies. While these aspects have become increasingly established in curricula, current educational requirements place a stronger emphasis on cross-curricular competencies, systematic problem-solving, and the ability to understand, analyse, and design digital processes (Bocconi et al., 2022). This is particularly relevant in primary education, especially in contexts where computer science and digital education are implemented not only as separate subjects but also in an interdisciplinary manner, as reflected in the Swiss Curriculum 21 and its module "Media Education and Informatics" (<https://www.lehrplan21.ch/>). Therefore, elementary school students should instead develop a deeper understanding of computational concepts and their application in meaningful contexts. Within this context, Computational Thinking (CT) has emerged as a fundamental, universally applicable 21st-century competency that is essential for understanding computational processes, abstracting information, and systematic problem-solving (Choi and Choi, 2024; Monge Roffarello and Sáenz, 2025). As education around the world increasingly advocate for the integration of CT into K-12 curricula starting in the earliest grades, elementary schools face the challenge of implementing these guidelines in their daily teaching practices (Dagiené et al., 2022).

However, a significant barrier to the effective integration of CT in primary education is the insufficient preparation of teachers, which extends far beyond a mere absence of technical skills (Ortuño Meseguer and Serrano, 2024). Whilst many primary school educators experience feelings of anxiety or intimidation when confronted with the subject of coding or computing, they also face substantial pedagogical challenges. A prevalent issue is the lack of Technological Pedagogical Content Knowledge (TPACK); teachers often struggle with how to design effective learning scenarios and meaningfully integrate CT into authentic, cross-curricular situations rather than merely teaching it in isolation. In addition, educators frequently concentrate their professional development to an excessive extent on technical programming environments or robotics, thus overlooking unplugged activities and more extensive didactic strategies that encourage creativity (Ortuño Meseguer and Serrano, 2024). Thus, achieving meaningful computational empowerment remains a major challenge (Dagiené et al., 2022).

Within this context one pillar of the Erasmus+ project "ComeThinkAgain" (COMputational and Entrepreneurship THINKing And Green Transitions, co-funded by the European Union, ERASMUS-EDU-2023-PI-ALL-INNO, and

Movetia: Swiss contribution) is focussed on CT. The project is driven by the need to equip teachers and VET trainers with interdisciplinary competencies. This objective is pursued by developing a Certification-based Education Training System (CETS) that integrates Computational Thinking (CT), Entrepreneurship Education (EE), and Green Skills (GS). To address the specific challenges encountered in primary teacher education, one of the first eight modules was designed as a 1-ECTS micro-module on CT foundations for in-service primary school teachers.

To reduce the barriers to entry and reduce anxieties related to computer science, the module employs, in addition to other methods, a Game-Based Learning (GBL) approach, adopting a playful methodology, which will be employed in Section 2. In addition, the present paper delineates preliminary pilot results of this module in Section 3. The study is guided by the following research question: **How can a playful, game-based and unplugged professional development module support primary school teachers in developing their understanding of CT and their confidence in applying CT across the curriculum?** Therefore, rather than aiming to establish generalisable effects, the study focuses on initial qualitative insights into how primary teachers experienced playful, unplugged, and game-based approaches to CT and how they transferred these approaches into their own cross-curricular lesson designs. Results are described in Section 4 and discussed in Section 5. The paper concludes with a conclusion and outlook.

2. Literature Review

This section introduces the theoretical framework of the study by outlining key dimensions of CT, challenges in primary teacher education, and the potential of game-based, unplugged, and maker-oriented approaches for teaching CT.

2.1 The Essence and Core Dimensions of Computational Thinking

The concept of CT was first introduced by Seymour Papert's (1980) constructionist learning theory, which posited that knowledge is most effectively constructed when learners actively engage in creating meaningful artefacts (Papert, 1980). The term was subsequently popularised by Wing (2006), who defined CT as a thinking process based on computational concepts that enable problem-solving and system design (Wing, 2006). CT is not about thinking *like a computer*, but rather *like a computer scientist*—formulating problems in a way that allows a computational agent (human or machine) to execute the solution.

Recent literature highlights that CT encompasses a spectrum of cognitive abilities that are applicable across various fields (Mohamed et al., 2024). The core dimensions of CT generally include the following (Hsu et al., 2023; Mohamed et al., 2024):

- Decomposition: the process of dividing complex problems into more manageable parts
- Abstraction: the focus on relevant information while ignoring unnecessary details
- Pattern recognition: the identification of regularities to predict outcomes
- Algorithmic thinking: the design of step-by-step instructions

Building on these foundations, further concepts support the development and implementation of solutions. Furthermore, the acquisition of these competencies is conducive to the development of transferable skills that are of significant value in a data-driven world. These include logical reasoning, critical thinking and systematic problem-solving (Mohamed et al., 2024).

2.2 Challenges Encountered in Primary Teacher Education for CT

Notwithstanding the acknowledged importance of CT, its implementation in primary education is frequently impeded by the so-called "teacher training issue" (Monge Roffarello and Sáenz, 2025). A systematic review conducted by Ortuño Meseguer and Serrano (2024) has revealed that a significant proportion of primary school teachers endeavour to integrate CT into their classrooms without possessing the requisite pedagogical competencies or prior training. This deficiency frequently gives rise to incorrect assumptions. For instance, Dagiené et al. (2022) found that the majority of primary school teachers falsely associate CT exclusively with the use of computers and specific digital tools, overlooking its broader cognitive and problem-solving applications. A study conducted in Zurich, Switzerland, with pre-service teachers reaches similar conclusions (Spieler et al., 2022). To integrate CT into authentic, cross-curricular learning scenarios, teacher education must foster what has been described as «Computational Empowerment» (Dindler et al., 2022). This concept expands the generic definition of CT and adds a broader goal of developing an understanding of the effects of digital technology on life and society. This is achieved by providing a framework to analyse and reflect technology.

2.3 Game-based Learning, Unplugged Activities, and Maker Education as Pedagogical Catalysts

To overcome the intimidation associated with computer science, literature increasingly advocates for the use of Game-Based Learning (GBL) and "unplugged" (offline) activities. GBL employs the engaging elements of games as pedagogical tools to stimulate active learning, motivation, and persistence in problem-solving (Mohamed et al., 2024; Giannakoulas and Xinogalos, 2024). Educational games have been shown to minimise the initial cognitive load by visually representing abstract concepts (Choi and Choi, 2024), thus allowing learners to focus on the semantics of computational logic rather than getting frustrated by syntax errors. Moreover, GBL necessitates adherence to a set of rules, the development of strategies, and the execution of tasks in a sequential manner, thereby implicitly fostering algorithmic thinking and debugging skills (Mohamed et al., 2024). Through the analysis of the rules and mechanics of well-known video games, educators can readily comprehend abstract concepts such as decomposition and abstraction (Holenko Dlab et al., 2020).

However, before transitioning to these digital "plugged" exercises, literature suggests that young learners, and their teachers, benefit immensely from building a conceptual foundation through unplugged tasks. As posited by Del Olmo-Muñoz et al. (2020), students have been shown to achieve superior learning outcomes when they initially engage with CT concepts via offline materials, such as unplugged game-based learning activities or logic puzzles, prior to transitioning into digital environments. The **Bebras Challenge**, **code.org** and **CS Unplugged** are three frameworks that support these tasks, rendering computational thinking concepts comprehensible and action-oriented without the barrier of technological complexity (Futschek and Moschitz, 2011).

To establish a connection between abstract logic and physical action, the utilisation of **Educational Robots and Visual Programming Languages (VPLs)** is a common practice. The utilisation of tangible tools such as Bee-Bots, Ozobots, and Thymio facilitates children's engagement with the concept of sequencing through physical interaction. This approach enables them to transition from rudimentary trial-and-error methods to a more systematic and strategic thinking process (Chevalier et al., 2020). VPLs are frequently associated with environments such as Scratch, which employ visual metaphors to represent programming primitives (Tsai, 2019).

In addition, **storytelling and maker education** provide essential context and creative outlets for young students. McNett (2016) hypothesises that the active engagement of the brain is facilitated by the contextualisation of principles within relatable narratives, rendering information more accessible and memorable. Conversely, the making approach integrates CT with hands-on experimentation, wherein students utilise microcontrollers such as BBC micro:bit to construct tangible artefacts (Kalogiannakis et al., 2021). This methodology fosters an iterative design approach, allowing students who may struggle with abstract problems to find success through the construction and modification of tangible objects.

When teachers engage with CT through physical challenges, familiar games, stories, robotics, or maker activities, they can recognise CT as something they may already use intuitively in their teaching (Dagienė et al., 2022). Combining unplugged modelling with game analysis and hands-on activities can therefore provide a practical framework for building confidence and supporting CT across different subjects.

3. The "Computational Thinking" Micro-module for Primary Education

The CT micro-module for primary education was developed by the Zurich University of Teacher Education (PHZH), Switzerland, and conducted between October 2025 and April 2026. It was designed as a blended learning experience, guiding teachers through eight progressive phases from theoretical foundations to practical classroom application. The structure of the micro-module is outlined as follows: The course starts with **Phase 1**, which incorporates a pre-survey to evaluate the participants' initial CT knowledge and self-efficacy. This is followed by a four-hour distance learning segment in which the participants acquire the theoretical foundations of CT by reading literature, answering reflective questions and a gaming task. In this activity, the teachers first actively played *Super Mario Bros* and then analytically deconstructed its game mechanics. They identified the static and dynamic elements of the game, mapped their relationships, and created visual models of the character's abilities and the system's rules. Proceeding to **Phase 2**, the teachers partake in an in-person workshop spanning 3.5 hours, centred on the practical application of these foundations through dedicated stations focussed on the top four CT concepts: decomposition, abstraction, pattern recognition, and algorithmic thinking. A central tenet of this approach is the GBL strategy, wherein educators engage in different unplugged and plugged games, e.g., code.org, Bebras challenges or "brain games". By actively engaging with puzzles, card tricks, building blocks, and simple robots, they experience firsthand how abstract logic can be made fun, accessible, and motivating. In **Phase 3**, a six-hour practice transfer period allows teachers to reflect on and contextualize their new knowledge and make own stations on further concepts (including e.g., generalization,

debugging, or programming concepts such loops, conditions), which is further deepened in a one-hour online session (**Phase 4**) to present their work. Another online activity was an "algorithmic walk," where step-by-step instructions are tested in real-world environments, in this case with Google maps. **Phase 5** was again a self-distance learning phase. Participants completed a reading task on CT methods, with a shift towards understanding CT as a process that can also be applied to everyday problems, such as planning a birthday party. **Phase 6**, again conducted on-site, shifted the focus towards cross-curricular integration, under the slogan "Hack the Curriculum". During this phase, participants explore CS Unplugged activities, such integrating CT into subjects like Science and Society (NMG) by modelling simplified food webs and simulating ecosystem changes, or into textile design using digital tools like Embroidery Designer (Spieler et al., 2024). **Phase 7** involves a second, six-hour practice transfer in which educators are tasked with the autonomous design of a progressive, cross-curricular CT lesson sequence of approximately 90 minutes, aimed at non-CS subjects and including formative or summative assessments. **Phase 8** of the programme culminates in a certification event that is designed to resemble a "Bazaar" or marketplace. This is followed by a post-survey comprising self-assessments and knowledge questions and a final certification test. Impressions from the module are illustrated in Figure 1.



Figure 1: Examples from the CT module illustrating how abstract CT concepts were translated into tangible and playful learning activities: a) deconstructing a familiar game to identify rules, entities, actions, and dependencies; b) modelling decomposition through examples from Science and Society; c) using robotics and making activities to connect sequencing, testing, and debugging with physical artefacts; d) supporting abstraction with LEGO by reducing complex everyday problems to relevant features and relations. Table 1 summarises the mapping between selected module activities, the CT concepts addressed, and their pedagogical function within the professional development setting

Table 1: Mapping module activities to CT/GBL-concepts.

Activity	CT concepts addressed	Game-based or unplugged element	Pedagogical function
Super Mario Bros. analysis	Decomposition, abstraction, pattern recognition, algorithmic thinking	Game-based deconstruction of rules, entities, actions, and feedback	Supports teachers in identifying CT concepts in familiar game systems
Bebras and logic puzzles	Pattern recognition, abstraction, algorithmic thinking	Unplugged problem-solving challenges	Makes CT accessible without programming
Card tricks and brain games	Pattern recognition, algorithms, debugging	Playful rule-based tasks	Encourages systematic reasoning and hypothesis testing
Movement-based activities and algorithmic walk	Algorithmic thinking, sequencing, debugging	Physical, unplugged enactment of instructions	Highlights the need for precision and iterative refinement
Robotics activities	Sequencing, debugging, algorithmic thinking	Tangible interaction with educational robots	Bridges unplugged CT and basic programming through immediate feedback
Modelling food webs or route planning	Decomposition, abstraction, algorithmic thinking	Cross-curricular modelling task	Shows how CT can support subject-specific reasoning beyond computer science
Teacher-designed lesson artefacts	Transfer of multiple CT concepts	Adaptation into classroom activities	Demonstrates how participants translated CT concepts into their own teaching contexts

4. Methods

4.1 Participants

The pilot course initially started with a cohort of eight in-service primary school teachers. Following the first session, two teachers withdrew from the program, citing the estimated 30-hour workload as too demanding to manage alongside their regular teaching responsibilities. A further participant withdrew during the second practical transfer phase (Phase 7) also due to a heavy working load. Consequently, five educators, four females and one male, successfully completed the module and the final certification. The teachers were responsible for classes across Grades 2 to 6 from the canton of Zurich/Switzerland, thereby covering a broad range of primary school levels. One of the participants held a specialized role as a gifted education teacher, providing additional expertise in working with high-achieving students.

4.2 Data Collection and Evaluation

To assess the module's impact on the teachers' understanding of CT and their confidence in applying it, a qualitative research approach was adopted. Immediately following the certification process, focus group interviews were conducted. The available participants were divided into two focus groups, two teachers in each group. One absent participant was interviewed separately via Microsoft Teams.

The qualitative data were analysed using a thematic analysis approach. Although the interview sessions were audio-recorded, the analysis was primarily based on written protocols during the conversations rather than on full verbatim transcriptions. This decision was made in the context of the broader ComeThinkAgain pilot phase, which included more than ten pilots of different modules across several European partner institutions. To support comparability across these pilots, a shared documentation approach based on structured notes and protocols was used. First, the written protocols were read repeatedly to identify recurring statements related to teachers' understanding of CT, perceived confidence, perceived usefulness of unplugged and game-based activities, and transfer into classroom practice. Initial codes were then grouped into broader themes. In addition to the interview data, the practical teaching examples and lesson sequences designed by the educators during the transfer phase were collected. These artefacts served as supplementary evidence of participants' applied learning and were used to examine whether the reported perceptions were reflected in their lesson designs. Pre- and post-surveys were included as part of the module design and the broader cross-module evaluation of the ComeThinkAgain project. As these data had not yet been fully analysed at the time of writing, the present paper focuses on qualitative feedback and teacher-designed artefacts from this CT pilot.

5. Results

This section presents the qualitative findings from the pilot implementation of the CT micro-module. The results draw on focus group interviews, one additional individual interview, and teacher-designed lesson artefacts from the transfer phase.

5.1 Group Interviews

A key finding is that participants did not perceive CT as entirely new, but rather as a conceptual reframing of practices they were already applying. One participant said: "how ubiquitous it is and how intuitively we actually do many things without having a name for them" (participant 1, module CT), while another one added: "I now have the feeling that everything is computational thinking. Basically, you can solve anything using some method or a combination of them" (participant 2, module CT). The module helped them to recognize and name these practices more explicitly, increasing their awareness of CT in their teaching. The most valued aspect by all participants was peer exchange, underscoring the importance of collaborative reflection. Here, one participant specified this by saying: *"exchanging ideas, looking at what others have done"* (participant 2, course CT).

5.1.1 Shifting conceptions: from programming to playful problem solving

Initial feedback from the focus group revealed a prevalent misconception regarding CT. Participants predominantly associated CT with traditional computer science and programming. For instance, one participant explicitly expected to learn programming and how to use software tools, expressing initial disappointment when the module focussed on other methodologies. Another participant admitted to feeling insecure about programming but chose the module to better understand underlying processes and terminology.

However, the introduction of unplugged and game-based approaches successfully dismantled these apprehensions. Participants' comments indicate a shift in how they conceptualised CT, from a primarily

programming-related activity towards a broader problem-solving perspective. They recognized that "CT is everywhere" and that they had already been utilising CT methods intuitively in their teaching, also without the formal terminology. One teacher described this perspective as putting on "glasses" to see everyday processes through a computational lens. The teachers highlighted that many existing educational materials already contain CT elements, but lack the explicit labelling needed to teach them consciously.

5.1.2 The impact of unplugged and game-based learning elements

The module's pedagogical design, leveraging game mechanics and unplugged activities, proved highly effective for this target group. Participants emphasized that the interactive components, specifically videos and games, were instrumental to their learning process. The hands-on, unplugged stations were frequently cited as a motivational highlight. Participants praised the on-site "bazaar" featuring various interactive stations to try out and explore. These tactile, gamified experiences were crucial for translating the abstract concepts of CT into tangible, relatable experiences.

Furthermore, the analytical deconstruction of video games prompted a desire for deeper engagement. Specifically, one participant noted that the Super Mario Bros game analysis example was a focal point that strongly warranted further collaborative discussion among the group.

5.1.3 Teacher confidence and classroom implementation

A primary goal of the module was to build confidence among educators lacking a computer science background. The qualitative feedback suggests that the module contributed to increased perceived confidence among the participating teachers. Participants were "pleasantly surprised" by how accessible the subject matter was.

The playful and hands-on nature of the module directly translated into classroom readiness. Teachers reported that they are already applying the newly acquired CT concepts in their daily practice. Instead of teaching programming alone, they use CT principles to "slow down" activities, helping pupils become consciously aware of individual steps and algorithms in everyday problem-solving. Overall, the participants found the practical, open-book certification and the mix of self-paced online learning with mandatory, interactive in-person sessions to be highly motivating and highly applicable to their diverse classroom needs.

5.2 Teacher Projects During the Transfer Phase (Phase 7)

The participating teachers implemented various classroom projects integrating CT across subjects, see Table 2.

Table 2: Teacher-designed classroom activities and their CT-related transfer

Teacher project	Subject area	CT concepts	Game/playful element	Evidence of transfer
Story hacking	Language / literacy	decomposition, abstraction	narrative exploration	analyse story structure
Ball games	Physical education	pattern recognition, algorithms	movement-based game design	students design own rule-based games
Trash route planning	NMG / sustainability	decomposition, algorithmic thinking, debugging	map-based challenge	iterative route optimisation
Raccoon / Blue-Bot	Early programming / storytelling	sequencing, debugging	narrative robot mission	story used to motivate algorithmic navigation
Designing and analysing spatial structures using 2D blueprints	Mathematics, geometry	decomposition, abstraction	simulation, physical sandbox	cross-subject integration

Some of the artefacts are illustrated in Figure 2. These artefacts provide evidence of transfer, as participants did not merely reproduce course activities but adapted CT principles to their own subjects and grade levels.



Figure 2: Teacher-designed classroom activities from the transfer phase. For example: route planning and debugging with a simplified map, narrative-based robot navigation, pattern recognition in physical education, and algorithmic sequencing through movement-based tasks

6. Discussion

The findings suggest that playful, unplugged, and game-based approaches can provide a low-threshold entry point for introducing CT to primary school teachers without a formal computer science background. Rather than presenting CT primarily as programming, the module framed it as a way of structuring, analysing, and modelling problems. This appeared to help participants recognise CT in familiar classroom practices and reduce initial barriers associated with computing.

The deconstruction of familiar games was particularly useful for making abstract CT concepts more tangible. By analysing rules, actions, conditions, and feedback structures, participants connected concepts such as decomposition, abstraction, pattern recognition, and algorithmic thinking to experiences they already understood. Similarly, hands-on stations, movement-based activities, puzzles, robotics, and classroom-oriented examples supported conceptual confidence before moving towards more technical forms of implementation.

The teacher-designed artefacts provide initial evidence of classroom transfer. Participants did not simply reproduce course activities, but adapted CT principles to language, physical education, sustainability-related tasks, route planning, geometry, and robot-based storytelling. This indicates that playful and unplugged approaches may help teachers identify CT opportunities beyond computer science and integrate them into cross-curricular primary education.

Limitations: The study is limited by its small sample, as only five teachers completed the full module. In addition, the qualitative analysis was based primarily on written protocols rather than full transcripts. The pilot also revealed practical challenges concerning workload, as three participants withdrew during the course. Future implementations should therefore consider shorter formats, such as one-day on-site workshops, to introduce selected elements of the module while better aligning with teachers' available time.

7. Conclusion and Outlook

The 1-ECTS CT micro-module of the "ComeThinkAgain" project offers a highly promising blueprint for training primary school teachers in CT. The project has entered its second development phase, which focuses on the creation of specific "intersection modules." In this phase, six additional modules will be added, with their pilot phase starting Autumn 2026. A fundamental objective of these new modules is to synergistically combine the

three competence areas of CT, GS and EE. At PHZH the focus will be again on primary education within the intersection of all three competence pillars. This time, the focus will be on design thinking, sustainability, and microcontrollers. All five teachers from the CT module have already registered for this module.

Acknowledgement

Co-funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the EACEA. Neither the European Union nor the granting authority can be held responsible for them. The Swiss participation is co-funded by Movetia, the Swiss agency for exchange and mobility.

Ethics Declaration: Formal ethical clearance was not required for this study. Participation was voluntary, and all data were anonymised for reporting purposes.

AI Declaration: AI tools were used to support language editing, academic phrasing, and structural refinement. The authors reviewed and approved the final text and remain fully responsible for its content.

References

- Bocconi, S., Chiocciariello, A., Kampylis, P., Dagienė, V., Wastiau, P., Engelhardt, K., Earp, J., Horvath, M.A., Jasutė, E., Malagoli, C., Masiulionytė-Dagienė, V. and Stupurienė, G. (2022) *Reviewing Computational Thinking in Compulsory Education*. Edited by A. Inamorato Dos Santos, R. Cachia, N. Giannoutsou and Y. Punie. Luxembourg: Publications Office of the European Union. doi: 10.2760/126955.
- Chevalier, M., Giang, C., Piatti, A. and Mondada, F. (2020) 'Fostering computational thinking through educational robotics: A model for creative computational problem solving', *International Journal of STEM Education*, 7(1), 39. doi: 10.1186/s40594-020-00238-z.
- Choi, W.C. and Choi, I.C. (2024) 'The influence of CodeCombat on computational thinking in Python programming learning at primary school', in *Proceedings of the 2024 5th International Conference on Education Development and Studies (ICEDS '24)*. New York, NY: Association for Computing Machinery, pp. 26–32. doi: 10.1145/3669947.3669951.
- Dagienė, V., Jevsikova, T., Stupurienė, G. and Juškevičienė, A. (2022) 'Teaching computational thinking in primary schools: Worldwide trends and teachers' attitudes', *Computer Science and Information Systems*, 19(1), pp. 1–24. doi: 10.2298/CSIS201215033D.
- Del Olmo-Muñoz, J., Cózar-Gutiérrez, R. and González-Calero, J.A. (2020) 'Computational thinking through unplugged activities in early years of primary education', *Computers & Education*, 150, 103832. doi: 10.1016/j.compedu.2020.103832.
- Dindler, C., Iversen, O., Caspersen, M. and Smith, R. (2022) 'Computational empowerment', in *Computational Thinking Education in K–12: Artificial Intelligence Literacy and Physical Computing*. pp. 121–139. doi: 10.7551/mitpress/13375.003.0009.
- Futschek, G. and Moschitz, J. (2011) 'Learning algorithmic thinking with tangible objects eases transition to computer programming', in I. Kalaš and R.T. Mittermeir (eds.) *Informatics in Schools: Contributing to 21st Century Education*. Berlin: Springer, pp. 155–164. doi: 10.1007/978-3-642-24722-4_14.
- Giannakoulas, A. and Xinogalos, S. (2024) 'Studying the effects of educational games on cultivating computational thinking skills to primary school students: A systematic literature review', *Journal of Computers in Education*, 11(4), pp. 1283–1325. doi: 10.1007/s40692-023-00300-z.
- Holenko Dlab, M., Hoic-Bozic, N., Mezak, J. and Zunic, M. (2020) 'Supporting Croatian primary school teachers in designing game-based learning activities: A case study', in *Proceedings of the European Conference on Games Based Learning (ECGBL)*. Brighton, UK.
- Hsu, T.-C., Chang, Y.-S., Chen, M.-S., Tsai, I.-F. and Yu, C.-Y. (2023) 'A validity and reliability study of the formative model for the indicators of STEAM education creations', *Education and Information Technologies*, 28(7), pp. 8855–8878. doi: 10.1007/s10639-022-11412-x.
- Kalogiannakis, M., Tzagaraki, E. and Papadakis, S. (2021) 'A systematic review of the use of BBC Micro:bit in primary school', in *Proceedings of the International Conference. New Perspectives in Science Education 2021*. pp. 1–4. Available at: https://conference.pixel-online.net/library_scheda.php?id_abs=5036 (Accessed: 1 July 2026).
- McNett, G. (2016) 'Using stories to facilitate learning', *College Teaching*, 64(4), pp. 184–193. doi: 10.1080/87567555.2016.1189389.
- Mohamed, Z., Ubaidullah, N.H., Md Junus, N.W., Angamuthu, K.D. and Ahmad, A. (2024) 'Modelling computational thinking with game-based learning among primary school students', *International Journal of Evaluation and Research in Education (IJERE)*, 13(6), pp. 4115–4124. doi: 10.11591/ijere.v13i6.28395.
- Monge Roffarello, A. and Sáenz, J.P. (2025) 'Supporting teachers, engaging students: A collaborative model for K-12 computing education', *Entertainment Computing*, 54, 100937. doi: 10.1016/j.entcom.2025.100937.
- Ortuño Meseguer, G. and Serrano, J.L. (2024) 'Implementation and training of primary education teachers in computational thinking: A systematic review', *RIED-Revista Iberoamericana de Educación a Distancia*, 27(1), pp. 255–287. doi: 10.5944/ried.27.1.37572.
- Papert, S. (1980) *Mindstorms: Children, Computers, and Powerful Ideas*. Basic Books.

- Spieler, B., Gursch, S., Krnjic, V., Horneck, K. and Slany, W. (2024) 'Designing, coding and embroidering: A workflow for gender-sensitive and interdisciplinary teaching', *International Journal of Emerging Technologies in Learning (IJET)*, 19(06), pp. 79–104. doi: 10.3991/ijet.v19i06.48591.
- Spieler, B., Schifferle, T.M. and Berner, T. (2022) 'Beliefs and expectations of primary student teachers in informatics', in *Proceedings of the 17th Workshop in Primary and Secondary Computing Education (WiPSCE '22)*. New York, NY: Association for Computing Machinery, pp. 1–4. doi: 10.1145/3556787.3556868.
- Tsai, C.-Y. (2019) 'Improving students' understanding of basic programming concepts through visual programming language: The role of self-efficacy', *Computers in Human Behavior*, 95, pp. 224–232. doi: 10.1016/j.chb.2018.11.038.
- Wing, J.M. (2006) 'Computational thinking', *Communications of the ACM*, 49(3), pp. 33–35. doi: 10.1145/1118178.1118215.