This is an open access article under the
Creative Commons Attribution 4.0 International License

COMPUTATIONAL THINKING BEYOND CODING IN EARLY CHILDHOOD EDUCATION: REFRAMING CS EDUCATION FOR THE AGE OF AI

Stamatios Papadakis
University of Crete, Greece

Computational thinking (CT) is increasingly recognized globally as a fundamental skill for the 21st century, yet its implementation in computer science (CS) education often is still limited to coding activities (Bers, 2018). It is essential to reframe CT as a cognitive approach that supports computational inquiry, problem-solving, and ethical AI engagement, starting from early education using developmentally appropriate tools like ScratchJr (Papadakis, 2020). In the age of artificial intelligence and rapid technological advancements, CS education faces a critical need to evolve beyond traditional approaches. Computational thinking (CT), often associated narrowly with coding, should be reframed as a cognitive framework essential for understanding computational processes, algorithmic thinking, and AI systems (Wing, 2016). Research indicates a growing need for robust frameworks to assess STEM competences, particularly in the context of current educational challenges (Grönman et al., 2025). Countries such as Singapore, Finland, and the United States have integrated CT into CS curricula to foster analytical and problem-solving skills (Pokropek, 2024). The OECD and UNESCO frameworks emphasize CT for addressing global challenges while supporting computational literacy. This reframing aligns with the EU Digital Education Action Plan and ACM/IEEE Computer Science Curriculum Guidelines for K-12 Education, emphasizing the need for robust CS skills to sustain competitiveness while addressing skill gaps (European Commission, 2025).

CT extends beyond basic programming skills, supporting systematic problem-solving, algorithmic thinking, abstraction, and data literacy fundamental to computer science education (Kalogiannakis & Papadakis, 2019). In early childhood contexts, CT enables learners to grasp core CS concepts, such as sequencing, debugging, iterative logic, and basic data representation (Bers, 2018). Tools like ScratchJr provide young learners with accessible, hands-on opportunities to engage directly with foundational CS concepts without cognitive overload (Papadakis, 2022). This approach aligns with both the European Commission's *DigComp 2.2* framework—which calls explicitly for addressing emerging digital skill gaps—and the ACM/IEEECS curricular guidelines, which emphasize competency-based computing education to close the CS skills divide (ACM & IEEECS, 2021; Vuorikari et al., 2022) (see Table 1).

https://doi.org/10.33225/jbse/25.24.592

ISSN 1648–3898 /Print/
ISSN 2538–7138 /Online/

Table 1
Linking ScratchJr to EU Key Competences

Activity Example	Key Competence
Animate a morning routine	Digital Competence
Fix a broken animation	Learning to Learn, Problem Solving
Tell a story with characters	Literacy, Cultural Expression
Work in groups	Social and Civic Competences
Rebuild a sequence	Mathematical Competence

Computational literacy today requires more than coding proficiency; it involves the ability to use, question, and generate algorithmic solutions and understand data critically (TeachAI & CSTA, 2025). CT offers tools necessary for algorithm testing, data interpretation, and computational thinking, critical for engaging effectively with AI systems and digital information (Pokropek, 2024). The EU emphasizes the importance of enhancing digital competences and updating its competence frameworks, including AI and data literacy, as part of preparing critically thinking citizens for the AI era. The EU's Digital Education Action Plan underlines the significance of building robust digital and CS competencies alongside foundational literacies, aligning with the need to prepare critically thinking citizens for the AI era (European Commission, 2020, 2025).

Addressing the persistent gender gap in CS requires integrating CT within CS education to foster interest and confidence in computational methods among underrepresented groups (Evagorou et al., 2024). Evidence from interventions using ScratchJr in early and primary education shows that visual programming can lower entry barriers, foster computational problem-solving, and improve the understanding of computational concepts in inclusive environments. It does so by mitigating the negative effects of the hidden curriculum in reproducing gender stereotypes on the roles of different genders in ICT (Papadakis, 2018). Such approaches align with strategic goals for increasing participation in CS fields and fostering social equity (Papadakis et al., 2018; Yang & Bers, 2024).

As AI becomes embedded in computational research and daily life, students require a foundational understanding of algorithmic processes and data ethics to engage responsibly with technology (Colligan et al., 2025). CT offers a pedagogical entry point for teaching AI-related concepts, enabling learners to demystify black-box systems and understand the socio-technical implications of AI in computational contexts (Lee et al., 2025). In this context, integrating CT within CS education supports not only cognitive skill development but also ethical and reflective capacities essential for responsible digital citizenship.

Empirical findings reveal that CT interventions improve preschool children's problem-solving skills and mathematical reasoning, offering concrete evidence to advocate for CT's early introduction (Yang et al., 2025). Future research should address ethical implications of AI integration in early CS education and explore strategies for overcoming barriers in low-resource contexts. Investigations into teacher preparedness, parental engagement, and equitable access to digital resources represent critical areas for broader, sustainable implementation of CT (Allglen, 2025).

Integrating CT beyond coding within CS education globally requires policies that support teacher training, curriculum redesign, and the adoption of age-appropriate technologies (Woo & Falloon, 2025). It is vital for educators, policymakers, and researchers to recognize CT as an essential pillar of computational literacy in the AI era.

References

ACM & IEEE-CS. (2021). *Computing Curricula 2020: Paradigms for global computing education* (CC2020). ACM.

Allglen. (2025). Jevons paradox: Why junior developers shouldn't fear AI. *Stackademic*. https://blog.stackademic.com/jevons-paradox-why-junior-developers-shouldnt-fear-ai-b7daf86c5590

Bers, M. U. (2018). *Coding as a playground: Programming and computational thinking in the early childhood classroom*. Routledge.

Colligan, P., Griffiths, M., & Cucuiat, V. (2025). *Why kids still need to learn to code in the age of AI: A position paper from the Raspberry Pi Foundation*. Raspberry Pi Foundation. https://www.raspberrypi.org

European Commission, Directorate-General for Education, Youth, Sport and Culture. (2020, September 30). *Digital Education Action Plan (2021–2027): Resetting education and training for the digital age*. European Commission. https://education.ec.europa.eu/focus-topics/digital-education/action-plan

https://doi.org/10.33225/jbse/25.24.592

ISSN 1648–3898 /Print/
ISSN 2538–7138 /Online/

European Commission, Directorate-General for Education, Youth, Sport and Culture. (2025, July 10). *Updating the European Digital Competence Framework*. European Commission. https://education.ec.europa.eu/focus-topics/digital-education/action-plan/updating-the-european-digital-competence-framework

Pokropek, A. (2024). *STEM competences, challenges, and measurements: A literature review* (G. Mazzeo-Ortolani, Z. Karpinski, & F. Biagi, Eds.). European Commission, Joint Research Centre. Publications Office of the European Union. https://data.europa.eu/doi/10.2760/9390011

European Commission. (2025). A STEM education strategic plan: Skills for competitiveness and innovation. https://education.ec.europa.eu/education-levels/school-education/basic-skills

Evagorou, M., Puig, B., Bayram, D., & Janeckova, H. (2024). *Addressing the gender gap in STEM education across educational levels: Executive summary* (NESET Report). Publications Office of the European Union. https://doi.org/10.2766/32315

Grönman, S., Lindfors, E., & Rönkkö, M.-L. (2025). Design thinking in early childhood education and care: A literature review and consideration from the perspective of young learners' craft, design, and technology education. *International Journal of Technology and Design Education, 35*, 857–877. https://doi.org/10.1007/s10798-024-09944-z

Kalogiannakis, M., & Papadakis, S. (2019). Pre-service kindergarten teacher's acceptance of "ScratchJr" as a tool for learning and teaching computational thinking and science education. *The Journal of Emergent Science (JES)*, *(15)*, 31–34.

Lee, H.-P. (H.), Sarkar, A., Tankelevitch, L., Drosos, I., Rintel, S., Banks, R., & Wilson, N. (2025). The impact of generative AI on critical thinking: Self-reported reductions in cognitive effort and confidence effects from a survey of knowledge workers. In *CHI Conference on Human Factors in Computing Systems (CHI '25)* (pp. 1–23). Association for Computing Machinery. https://doi.org/10.1145/3706598.3713778

Papadakis, S. (2018). Gender stereotypes in Greek computer science school textbooks. *International Journal of Teaching and Case Studies*, *9*(1), 48–71.

Papadakis, S. (2020). Apps to promote computational thinking concepts and coding skills in children of preschool and pre-primary school age. In *Mobile learning applications in early childhood education* (pp. 101–121). IGI Global Scientific Publishing.

Papadakis, S. (2022). Apps to promote computational thinking and coding skills to young age children: A pedagogical challenge for the 21st century learners. *Educational Process: International Journal (EDUPIJ)*, *11*(1), 7–13.

Papadakis, S., Tousia, C., & Polychronaki, K. (2018). Women in computer science. The case study of the Computer Science Department of the University of Crete, Greece. *International Journal of Teaching and Case Studies*, *9*(2), 142–151.

TeachAI & CSTA. (2025). *Guidance on the future of computer science education in an age of AI*. TeachAI. https://teachai.org/cs

Vuorikari, R., Kluzer, S., & Punie, Y. (2022). *DigComp 2.2: The digital competence framework for citizens – With new examples of knowledge, skills, and attitudes*. Publications Office of the European Union. https://data.europa.eu/doi/10.2760/115376

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33–35.

Woo, K., & Falloon, G. (2025). Integrating coding across the curriculum: A scoping review. *Computer Science Education, 35*(2), 216–237. https://doi.org/10.1080/08993408.2024.2344402

Yang, D., Yang, Z., & Bers, M. U. (2025). The efficacy of a computer science curriculum for early childhood: Evidence from a randomized controlled trial in K-2 classrooms. *Computer Science Education, 35*(2), 321–341. https://doi.org/10.1080/08993408.2023.2279198

Yang, Z., & Bers, M. (2024). Examining gender difference in the use of ScratchJr in a programming curriculum for first graders. *Computer Science Education, 34*(4), 864–885. https://doi.org/10.1080/08993408.2023.2224135

Received: *July 10, 2025* Revised: *July 20, 2025* Accepted: *August 01, 2025*

Cite as: Papadakis, S. (2025). Computational thinking beyond coding in early childhood education: Reframing CS education for the age of AI. *Journal of Baltic Science Education*, *24*(4), 592–594. https://doi.org/10.33225/jbse/25.24.592



Stamatios Papadakis PhD, Assistant Professor in Educational Technology, Department of Preschool Education, University of Crete, Gallos Campus, Rethymno 74100, Greece.
E-mail: stpapadakis@uoc.gr
ORCID: https://orcid.org/0000-0003-3184-1147