frontiers | Frontiers in Education

TYPE Editorial
PUBLISHED 14 July 2023
DOI 10.3389/feduc.2023.1248509

Check for updates

OPEN ACCESS

EDITED AND REVIEWED BY
Eileen Scanlon,
The Open University, United Kingdom

*CORRESPONDENCE
Kati Mäkitalo
✉ kati.makitalo@oulu.fi

RECEIVED 27 June 2023
ACCEPTED 05 July 2023
PUBLISHED 14 July 2023

CITATION
Mäkitalo K, Laru J and Dagienė V (2023) Editorial: Looking ahead: computational thinking in K12 education. *Front. Educ.* 8:1248509. doi: 10.3389/feduc.2023.1248509

Editorial: Looking ahead: computational thinking in K12 education

Kati Mäkitalo[1]*, Jari Laru[1] and Valentina Dagienė[2]

[1]Research Unit of Learning Processes and Learning, Faculty of Education and Psychology, University of Oulu, Oulu, Finland, [2]Institute of Data Science and Digital Technologies, Faculty of Mathematics and Informatics, Vilnius University, Vilnius, Lithuania

KEYWORDS

computational thinking, curriculum, K12 education, teacher, student

Editorial on the Research Topic
Looking ahead: computational thinking in K12 education

1. Introduction

This Research Topic focuses on computational thinking (CT) and its integration into curricula and existing educational practices. The term informatics is commonly used to refer to Computer Science (CS) and computing to describe the disciplines that study and develop theories, models, and applications of information technology. CT shares elements with the principles and concepts of these disciplines, as they are built upon similar foundations, such as algorithms, sequences and parallelization, control and automation, and data processing, representation, and recognition (Wing, 2006; Denning and Tedre, 2019). Broadly speaking, CT refers to an approach of solving complex problems by applying a set of concepts, practices, and dispositions that are fundamental to CS. In CS education, CT, a versatile concept that can be applied to different contexts, may be used as an aid to understand the principles and concepts of CS. In its broadened definition, different learning strategies can be considered within the scope of CT. For example, strategies such as problem-based learning, collaborative learning, project-based learning, game-based learning, scaffolding, or storytelling have been identified in literature reviews (Hsu et al., 2018; Veerasamy et al., 2021).

The articles in this Research Topic explore some challenges and potential promises of CT by sharing suitable approaches to integrate CT into K12 education. The authors of these articles describe and elaborate on approaches for tackling the various challenges, such as applying CT and grasping abstract concepts of CT, identifying CT skills, and integrating CT into curricula.

Fagerlund et al. highlight the role of teachers' and students' programming motivation in CT. Based on the results, the gender issue is raised, where boys were found to be more motivated for programming than girls. There is a need to provide positive CT experiences for girls to increase their programming motivation. Regarding teachers, the same phenomenon is observed for female teachers. Positive CT experiences relevant to their subject, with the objective of increasing programming motivation, especially for teachers who have less programming teaching experience or interest in the topic, are required. These results indicates that for teachers with less CT teaching experience, the training provided should especially focus on the subject they are teaching as well as on specific objectives to increase their motivation to integrate CT into curricula.

In the article entitled "Grades 7-12 teachers' perception of computational thinking for mathematics and technology", the authors Humble and Mozelius investigate teachers' understanding and perception of CT as a concept, and how teachers integrate CT into their teaching activities. Data include essay assignments from teachers in mathematics and technology in grades 7–12. The results indicated that teachers perceive CT's potential in fostering teaching and learning activities in mathematics and technology as well as in other STEM subjects. Activities with CT can be seen as strengthening the subjects when adopting new practices and emphasizing old practices. However, to be successfully implemented into teaching and learning activities, CT should be concrete and profound instead of abstractive and superficial.

The article by Oyelere et al. introduces a pedagogical evaluation framework to support educators in assessing technological instructional tools. Five pedagogical indicators for teaching and learning CT are identified; technology, pedagogical approaches, assessment techniques, data aspects, and teachers' professional development. The authors analyzed three instructional tools and secondary documents by utilizing the initial computational thinking of pedagogical assessment framework (CT-PAF). As the results show, the CT-PAF assessment tool is suitable for educators to evaluate the different technological learning tools in terms of pedagogical impact and outcome. Initial assessment of the framework highlights the shortcomings of the technological learning tools, such as data analytics and security, privacy, and ethics. The authors conclude that CT-PAF provides a robust support for teachers in decision-making regarding instructional tools which would enhance the learning outcomes in a K12 context.

Lindberg and Öberg introduce a study where a discursive analysis of policy documents and curricula was adapted to capture "the outworn dichotomy between the cultures", as the authors state in the article. The authors acknowledge differences between the disciplines (computer sciences and humanities) that influence curriculum discourses, national digital policy, and practices. Lindberg and Öberg question the effects of literacy, especially writing and reading skills nowadays and in the future, on current technological developments. Their vision for literacy includes the transformation from writing and reading text to writing and reading code. Based on results from the Swedish curricula, document programming is treated as specialized knowledge, which is part of mathematics and technology rather than other subjects such as humanities and social sciences. Their suggestion is to implement reading and writing code as digital literacy in education.

The contributions to this Research Topic show highlight critical issues that should be acknowledged both in scientific fields and practice. There is a need to widen our understanding of CT and to include discussions in our scientific contributions in both fields. As the articles showcase, CT is seen as programming and is part of few subjects in schools, which limits pupils' and teachers' understanding of CT. This narrow perspective limits our opportunities to better understand the possibilities of using technology and even the design of technology for our own use. CT includes not only programming skills but it is a part of our digital literacy and includes many other skills and knowledge than codes, such as problem solving, critical and logical thinking skills, and creativity. The integration of different disciplines is needed for exploring the meaningful use of technology in learning activities and outcomes in educational environments. To overcome these obstacles, more long-term cooperation is needed in order to influence the development of curricula as well as existing educational practices for different disciplines, considering the circumstances. In addition, it is necessary to adjust to a certain way of thinking. We should aim to well educate the younger generation; we have a responsibility to equip everyone, regardless of gender or socioeconomical background, with certain skills and knowledge that will enable them to function actively in an increasingly digitalized world.

Author contributions

All authors contributed to the article and approved the submitted version.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Denning, P. J. and Tedre, M. (2019). *Computational thinking.* Cambridge, MA: MIT Press.

Hsu, T.-C., Chang, S.-C. and Hung, Y.-T. (2018). How to learn and how to teach computational thinking: Suggestions based on a review of the literature. *Comput. Educ.* 126, 296–310. doi: 10.1016/j.compedu.2018.07.004

Veerasamy, A. K., Larsson, P., Apiola, M.-V., D'Souza, D., Laakso, M.-J. (2021). "Pedagogical approaches in computational thinking-integrated steam learning settings: a literature review," in *Intelligent Computing. Lecture Notes in Networks and Systems*, Arai, K. (ed.). Cham: Springer. doi: 10.1007/978-3-030-80129-8_27

Wing. J. M. (2006). Computational thinking. *Commun. ACM.* 49, 33–35. doi: 10.1145/1118178.1118215