Journal of Pedagogical Research
Volume 7, Issue 2, 2022
https://doi.org/10.33902/JPR.202313362

Introduction to the special issue on "Computational thinking and mathematics teaching and learning"

Guest Editors
Max Stephens[1] and Chantal Buteau[2]

[1]*The University of Melbourne, Australia (ORCID: 0000-0001-9744-4027)*
[2]*Brock University, Canada (ORCID: 0000-0001-9401-7011)*

We have been witnessing in the last decade an emerging and growing international conversation about computational thinking (CT) in education in response to the rapidly evolving technology development. At the core, many argue that all students in today's technology-based world should now develop an understanding of the functioning basics of digital technologies (what an algorithm is, what AI is, etc.) and that they should also be equipped to participate in solving problems with them – e.g., through creating a computer program (Modeste et al., in press). This newer need has led to different educational reforms (curriculum revisions) and changes in primary and secondary school classrooms, that integrate CT, in countries worldwide (Dagienė et al., 2019; Bocconi et al., 2016). Of importance, this call to promote CT in contemporary education has been reflected since 2021 by the assessment of students' CT by PISA, the Program for International Student Assessment (OECD 2018).

There has been long-standing and recent sound research (e.g. Leron & Dubinsky, 1995; Noss & Hoyles, 1996; Weintrop et al., 2016; Benton et al., 2017; Gadanidis, 2017; to name only a few) that supports different issues at stake concerning the potential of CT integration to support students learning. In mathematics, this research dates back to the work of Papert (1980) with 'Logo', a programming language designed for children, with an approach of having students turn the computer into an "object-to-think-with" (i.e., a powerful tool). Nevertheless, incorporating CT in mathematics, or in other disciplines such as STEM, in the school classroom still remains not straightforward and generates many practical challenges, such as: how to prepare teachers to meaningfully integrate CT activities; how to evaluate students' CT; how to design or select a rich CT-integrated mathematics task; ways to help teachers understand how students' learning of mathematics may be affected by a CT integration; etc.

This Special Issue focusses on the CT integration in mathematics education and aims at providing a forum to discuss diverse issues related to the school mathematics curriculum and classroom teaching and learning. We invited papers that would provide insights into clarifying and explaining the international trends and their growing impact on the curriculum, classroom practices, and learning– in the compulsory years of schooling as well as in the senior high school years. In particular, this Special Issue aims at bringing contributions relevant to teachers, mathematics curriculum experts, and teacher educators who are engaged in or keenly interested in these issues, from a practical point of view.

Of the numerous proposals received, eight papers are included in this Special Issue. These papers focus mainly on three areas, the first one being **CT as an agent for national curriculum reforms and development**—Whitney-Smith discusses "examples of alternative approaches to addressing CT in national curricula for the compulsory years of schooling" and explains "how CT has been adopted as a driver for mathematics curriculum change in Australia."

The second area addressed by the authors is **CT as a newer need in teacher education**— Moon and colleagues describe a "novel method of introducing [K-12 mathematics and science pre-service teachers] to CT through a five-lesson module within the context of an existing… methods course" to provide them with firsthand experience to help develop their "intuitions around CT" and "the roles it can play in their math and science classrooms;" Broley and colleagues present a collaborative professional development initiative involving in-service and pre-service mathematics teachers planning, implementing, and reflecting on the implementation of coding-based mathematics activities with Grade 5–9 school students, resulting in benefits for all of those involved; and Mumcu and colleagues discuss the implementation by pre-service teachers of an unplugged computer science activity to integrate CT into real mathematics classrooms and how this benefitted the middle school students, and highlighted the need to integrate classroom management as part of CT-related teacher education.

And the third main area is **CT as a disrupter or agent in (enhancing) student learning**— Munasinghe and colleagues explore and argue for the incorporation of unplugged activities to support school students' learning of programming, including in the context of mathematics education; Tupouniua discusses students' algorithmic thinking by presenting "three illustrative cases of emergent challenges evident as students grapple with the process of creating an algorithm" in three different mathematical algorithmatizing tasks, and ends with some practical pedagogical suggestions to address such challenges; Møller and colleagues present a study in which 12-15 year old children collaborated with their parents, in a non-formal out-of-school context, to "learn about mathematics and [CT] through a series of playful educational tasks with an educational robot" in order to examine "the STEM learning potential and obstacles" and parents-children relations in such a context; and Kaup and colleagues present a quantitative study aiming at examining whether CT-integrated interventions in a primary mathematics classroom related to the areas of a) number knowledge and arithmetic, b) algebra, and c) geometry, may positively impact students' performance.

Interestingly, the eight papers of this Special Issue stress the diversity of CT integration in compulsory education undertaken by different jurisdictions. We thus have asked the authors to separately provide a very brief overview of integration approaches in their respective jurisdictions in order to give a quick synopsis to the reader of this Issue. In the following, we report on them, first about Turkey, and then in the order of the papers' appearance in this Issue:

- **In Turkey (Mumcu et al. paper)**, computer science education in K-12 is compulsory in the 5th and 6th grades and the 9th grades of Science High Schools. It is optional in other levels and institutions. The curriculum published in 2018 aims to enable students to "acquire and develop problem-solving and computational thinking skills" (Turkey, Ministry of National Education, 2018). CT is included in the "Information Technologies and Software" course curriculum for students at all levels, from primary to secondary school. CT activities are included in this course's "Problem Solving and Programming" unit. The expected time allocated to the "Problem Solving and Programming" unit constitutes 50% of the total time. It covers learning outcomes for problem-solving, algorithm creation, programming components, block-based programs, and logic.
- **In the United States (Moon et al. paper)**, the last decade has seen a steady increase in the presence of CT across K-12 schools. The term CT, along with closely associated concepts and practices, such as problem decomposition and developing algorithms, are increasingly being found in standards documents, disciplinary frameworks, and new classroom curricula. The result is mathematics, science, computer science, and a growing array of social science and humanities K-12 classrooms that are incorporating CT.
- **In the province of Ontario in Canada (Broley et al. paper),** the Ministry of Education recently revised its elementary and lower secondary mathematics curricula, and introduced a new collection of "coding" expectations as part of the Algebra strand in Gr.1–9 (2020, 2021). It is thus expected that all students are "to learn to use coding concepts and skills to solve problems across mathematical topics and to create computational representations of

mathematical situations, by reading, altering, writing, and executing code" (Broley et al., p.20). The Ministry grounds the integration of coding, or more broadly CT, as part of the transferable skills that "are in high demand in today's globally connected world, with its unprecedented advancements in technology" (Ontario Ministry of Education, 2020, p. 30). Other Canadian provinces have also started integrating CT in their compulsory curricula; e.g., in Alberta as part of their science curriculum or in New-Brunswick as part of their technology curriculum.

- **As for Australia (Whitney-Smith paper),** CT has been an essential component of the curriculum since 2015, explicitly developed within the learning areas of Technologies. The revised Australian Curriculum Version 9.0, published in 2022, provides new explicit content within the learning area of Mathematics for students to develop their CT skills with applied learning opportunities across the other dimensions of the curriculum. Artificial intelligence, quantum computing, machine learning and other rapidly emerging technologies, coupled with the digitalisation of everyday practices and the shift towards being a data driven society, have led to a new way of working, consuming, communicating, and learning. Hence the growing importance for all students to be able to think computationally. As the implementation of the Australian Curriculum is the responsibility of the school and curriculum authorities in the eight Australian States and Territories, they decide how and when the Australian Curriculum is implemented in their jurisdiction, choosing to adopt or adapt. The Victorian Curriculum F-10, for example, introduced explicit CT content in Mathematics in 2017.
- **In New Zealand (Tupouniua paper; Munasinghe et al. paper),** as a result of their 2018 curriculum revision in which digital technologies was given greater emphasis throughout the compulsory curriculum, CT was positioned as one of the two main focus learning areas within the subject of Technology. The primary goal of the CT learning area is for students from Year 1 to 13 to gradually develop their understanding of core programming concepts such as algorithmization, abstraction, decomposition, and data representation through age-appropriate programming environments and authentic contextualized tasks. More generally, the overarching aim of promoting CT is to enhance student proficiency in not only using, but also creating digital technologies.
- **In Denmark (Møller paper; Kaup et al. paper),** CT is not yet implemented in the compulsory school national curricula. Technology Comprehension was tested in 46 schools and evaluated under two implementation strategies, including CT. The ideas were intended to inform a future decision regarding mainstream implementation. In contrast to the first strategy, which considered Technology Comprehension a separate subject, the second strategy incorporated Technology Comprehension competencies and learning goals into existing subjects such as Danish, mathematics, arts, physics/chemistry, science, craft and design, and social studies. A newly drafted curriculum, including mathematics, supported both approaches.

These diverse approaches to integrate CT in education also point to different understandings of CT, and its connections to programming and coding technology. The reader is invited to pay attention to this as they read the issue. Furthermore, the discussion concerning CT in schools continues to evolve, and for the interested reader, three papers in the Special Issue chart this development in some depth: those by Whitney-Smith, Moon and colleagues, and Broley and colleagues. In summary, the papers in this Special Issue, with their different contexts and understanding of CT, help us gain insight into diverse practical issues and opportunities, such as: a) ways to incorporate CT into national curriculum documents; b) ways, opportunities, and challenges to integrating CT into teacher preparation and teacher professional learning; c) classroom applications on related pedagogical issues; and d) involvement of different stakeholders (parents, pre-service teachers, etc.) in children's CT learning. With the continuing increase of CT integration in mathematics and STEM classrooms, more research is needed to better understand

and inform emerging situations in curriculum development involving CT, teacher education, teacher professional learning, and implementation in schools, including for example:

- Interfaces between CT, computer science, and mathematics and their relationship to the school curricula;
- Good pedagogy practices for teaching algorithmics and data analytics in the senior high school mathematics curricula;
- Assessing CT in mathematics classrooms and in large-scale national and international assessments;
- Exploring and evaluating different models of teaching and learning of CT in mathematics settings; and
- The relevance of artificial intelligence (AI) in school education as an emerging feature and now unavoidable component related to CT.

We end by thanking the contributing authors and all of the reviewers. As we well know, their contributions are what made this Issue possible. A special thanks to the editor-in-chief of this journal for his confidence in us to produce this Special Issue and for his unfailing guidance and support throughout its development.

References

Benton, L., Hoyles, C., Kalas, I., & Noss, R. (2017). Bridging primary programming and mathematics: Some findings of design research in England. *Digital Experiences in Mathematics Education, 3*, 115-138. https://doi.org/10.1007/s40751-017-0028-x

Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., & Engelhardt, K. (2016). *Developing computational thinking in compulsory education: Implications for policy and practice (JRC Science for Policy Report)*. European Commission. https://doi.org/10.2791/792158

Dagienė, V., Jevsikova, T., & Stupurienė, G. (2019). Introducing informatics in primary education: Curriculum and teachers' perspectives. In S. Pozdnyakov & V. Dagiene (Eds.), *ISSEP 2019: Informatics in Schools. New Ideas in School Informatics. Lecture Notes in Computer Science* (Vol. 11913, pp. 83-94). Springer. https://doi.org/10.1007/978-3-030-33759-9_7

Gadanidis, G. (2017). Five affordances of computational thinking to support elementary mathematics education. *Journal of Computers in Mathematics and Science Teaching*, *36*(2), 143–151.

Leron, U., & Dubinsky, E. (1995). An abstract algebra story. *American Mathematical Monthly, 102*(3), 227-242. https://doi.org/10.1080/00029890.1995.11990563

Modeste, S., Broley, L., Buteau, C., Rafalsaka, M., & Stephens, M. (in press). Computational thinking and mathematics. In B. Pepin, G. Gueudet, & J. Choppin (Eds.), *Handbook of Digital (Curriculum) Resources in Mathematics Education*. Springer.

Noss, R., & Hoyles, C. (1996). *Windows on mathematical meanings: Learning cultures and computers*. Kluwer. https://doi.org/10.1007/978-94-009-1696-8

OECD. (2018). *PISA 2021 mathematics framework (draft)*. https://www.oecd.org/pisa/sitedocument/PISA-2021-mathematics-framework.pdf

Ontario Ministry of Education. (2020). *Ontario curriculum and resources: Elementary mathematics.* https://www.dcp.edu.gov.on.ca/en/curriculum/elementary-mathematics

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

Turkey, Ministry of National Education (2018). *Bilişim Teknolojileri ve Yazilim Dersi Öğretim Programi.* Ankara. http://mufredat.meb.gov.tr/Dosyalar/2018124103559587-Bili%C5%9Fim%20Teknolojileri%20ve%20Yaz%C4%B1l%C4%B1m%205-6.%20S%C4%B1n%C4%B1flar.pdf

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal for Science Education and Technology, 25*, 127–147. https://doi.org/10.1007/s10956-015-9581-5