Empowering students' mathematics problem-solving skills: The role of computational thinking

Yimei Zhang, McGill University; Tanya Chichekian, Université de Sherbrooke and Annie Savard, McGill University

Problem-solving is one of the prominent skills that fosters students' understanding and application of mathematical concepts and content knowledge (Hiebert, et al. 1996). Although many students often report experiencing difficulties with problem-solving in mathematics (Berch & Mazzocco 2007), due to its importance in daily life and understanding the world around us (Tambychik & Meerah, 2010), empowering students with problem-solving skills is and remains an important goal in education.

Computational thinking (CT) represents a universally applicable skill set, not just for computer scientists, but for everyone who would be eager to learn and use it (Wing, 2011). The thought process involved in computational thinking refers to formulating problems and deriving solutions similar to that carried out by an information-processing agent (Wing, 2011). Research in Mathematics Education shows that implementing Computational Thinking (CT) into the curriculum fosters students' use of their cognitive abilities, contributes to the development of problem-solving skills, and, ultimately, to meaningful learning (Weintrop, et al. 2016). Retrospectively, the process of problem-solving in mathematics is also conducive to the application of CT (Hambrusch et al. 2009; Jona et al. 2014).

This study explores the potential of CT within current mathematics curriculums at Elementary Level (especially in the cultivation of Mathematics Problem-solving skills) to find out the potency and traces of CT's application and awareness in current Mathematics Education. The following research questions guided out study:

1) How can CT skills be employed to facilitate students' understanding of mathematical concepts and procedures?

2) To what extent can CT and problem-solving skills contribute to a deeper approach (e.g., improve efficiency, accuracy for Mathematics problem solving) to teaching and learning in mathematics?

Our knowledge taxonomy for mathematics problem-solving practice and skills from CT's perspectives consists of four categories: Decomposition, Abstraction, Generalization, Debugging:

- **Problem Decomposition** is a way to think about problems, algorithms, processes in terms of their component parts. The parts can then be understood, solved, developed, and evaluated separately. This makes complex problems easier to solve, novel situations better understood, and large systems easier to design.

- **Abstraction** is a way to make problems or systems easier to think about by removing unnecessary complexity. A key part of it is in choosing a good representation of a system. Different representations make different things easy to do.

- **Generalization** is associated with identifying patterns, similarities, and connections, and exploiting those features. It is a way of quickly solving new problems based on previous solutions to problems and building on prior experience. Asking questions such as “Is this like a problem I’ve already solved?” and “How is it different?” is important here, as is the process of recognizing patterns both in the data being used and the processes/strategies. Most importantly, the pedagogical power of computational generalization comes not just from students using existing models/solutions, but also from enabling students to design, build, and assess models/solutions of their own.

- **Debugging** is the systematic application of analysis and evaluation using skills such as testing, tracing, and logical thinking to predict and verify outcomes.

Table 1 presents some practical examples for mathematics problem-solving integrating CT skills.

Table 1: Mathematics Problem-solving with CT

Categories of PS	Sub-categories	Examples & Explanations
Decomposition	Read and Understand Problems	• The core of the mathematics problems • Stages to solve the problems • What kind of questions are they asking? • How are the problems formulated?
	Analyze and Plan	• Breakdown questions into simple steps • Breakdown complex questions into simple stages, while identifying the familiar problems which can be solved by any method we have mastered at any stage.
Abstraction	Reduction	• Remove unnecessary information • Translate the problems into mathematics sentences
	Decision Making	• Decide what information should be included or omitted
	Relationships	• Identify mathematical relationships within diverse contexts
Generalization	Pattern	• Identify patterns and similarities in problem-solving • Create models to illustrate problems
	Transformation	• Transfer ideas from one problem to another

		• Derive characteristics and uses of similar patterns
Debugging	Confirm the answer and the process Test the model and solution	• Assess that the result is accurate • Assess whether the algorithmic solution is functioning efficiently • Solve complex questions • Apply the solutions / process in novel contexts

** Note that the sub-categories and Examples & Explanations of mathematics problem-solving are task specific.*

In this study, we observed and analyzed Mathematics Problem-Solving procedures at the elementary level to inform how CT ideas and practices are enacted. This included how CT can be leveraged to provide opportunities for students to apply their problem-solving skills efficiently and how these skills can be cultivated at different stages. Practically, findings from this study highlighted how CT can be displayed both in mathematics teaching and learning, as well as the challenges associated with the latter. It seems the ability to embrace CT is crucial for mathematics problem-solving, especially when planning to teach problem-solving. Practicing the application of CT in problem-solving learning situations in mathematics could potentially facilitate the transfer of these skills in other disciplines.

References:

Hiebert, J., Carpenter, T. P., Fennema, E., Fuson, K., Human, P., Murray, H., ... & Wearne, D. (1996). Problem solving as a basis for reform in curriculum and instruction: The case of mathematics. *Educational researcher*, 25(4), 12-21.

Berch, D. B., & Mazzocco, M. M. (2007). Why is math so hard for some children? *The nature and origins of mathematical learning difficulties and disabilities*. Paul H. Brookes Publishing Co.

Tambychik, T., & Meerah, T. S. M. (2010). Students' difficulties in mathematics problem-solving: What do they say?. *Procedia-Social and Behavioral Sciences*, 8, 142-151.

Wing, J. (2011). Research notebook: Computational thinking—What and why. *The link magazine*, 6, 20-23.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of science education and technology*, 25(1), 127-147.

Hambrusch, S., Hoffmann, C., Korb, J. T., Haugan, M., & Hosking, A. L. (2009). A multidisciplinary approach towards computational thinking for science majors. *ACM Sigcse Bulletin*, 41(1), 183-187.