Computational Thinking and Problem Solving

Valentina Postelnicu
Governors State University

*Keywords:* computational thinking, problem solving, mathematics curriculum

Computational thinking (CT) is a way of thinking about problems that allows their formulation and solution to be expressed using computational steps or algorithms that can be carried out by humans and/or computers (Aho, 2012; Li et al., 2020; Wing, 2006, 2008). Often complemented by mathematical thinking, CT incorporates algorithmic thinking and focuses on problem solving using computer science concepts (e.g., data structures, algorithms, various programming styles, logic, and computation).

The purpose of this paper is to illustrate problem solving using a CT approach and discuss the role of CT within the K-20 mathematics curriculum.

Two studies conducted by the author (Postelnicu, 2025; Postelnicu & Postelnicu, 2020) point out the core components of CT:

- *decomposition* - breaking the problems into manageable components or steps;
- *pattern recognition* - identifying patterns;
- *abstraction* - simplifying the problem by reducing it to a mathematical model or a visual representation;
- *algorithm design* - write a logical sequence of computational steps or an algorithm.

Participants in the studies were middle school and high school students (first study), and mathematics and computer science educators with various backgrounds, including university professors (second study, ongoing). From a mathematical perspective, the problem from the first study can be considered an optimization problem, while the problem from the second study uses recurrence relations and modular arithmetic. The solutions for the two problems highlight the interplay between CT and mathematical thinking approaches. Both approaches to problem solving may share features like *decomposition*, *pattern recognition, abstraction*, and *algorithm design*, but those features shape up differently for each approach. For example, in the CT approach, the *abstraction* may come about through a visual representation (physical model, simulation), while in in the mathematical thinking approach *abstraction* comes about through a mathematical model. The type of a*bstraction* affects the way *pattern recognition* and *algorithm design* take shape. In the CT approach, the concern for *algorithm* efficiency and computational constraints often lead to "good enough" solutions or *algorithms* that are based on heuristics and may lack formal theoretical justifications. In the mathematical thinking approach, solutions or *algorithms* are based on theory, and the aim is to construct exact solutions or *algorithms*, rigorously justified, whenever possible.

The role of CT within the K-20 mathematics curriculum will be discussed. Weintrop et al. (2016) point to the challenges faced by integrating CT and subject-matter curricula, including defining CT and its role in the context of our mathematics curriculum. The discussion is pertinent, given that CT is necessary for understanding data, algorithms, and digital technologies that permeate multiple aspects of our life (Kafai and Proctor (2022).

## References

Aho, A. V. (2012). Computation and computational thinking. *The Computer Journal*, *55*(7), 832–835. https://doi.org/10.1093/comjnl/bxs074

Kafai, Y. B., & Proctor, C. (2022). A Revaluation of Computational Thinking in K–12 Education: Moving Toward Computational Literacies. *Educational Researcher*, *51*(2), 146–151. https://doi.org/10.3102/0013189x211057904

Li, Y., Schoenfeld, A.H., diSessa, A.A., Graesser, A.C., Benson, L.C., English, L.D., & Duschl, R.A. (2020). Computational Thinking Is More about Thinking than Computing. *Journal for Stem Education Research*, *3*, 1–18. https://doi.org/10.1007/s41979-020-00030-2

Postelnicu, V. (2025, February 4-8). Using computational thinking to solve the Josephus problem: Insights from participants with different educational backgrounds [Paper presentation]. 14th Congress of the European Society for Research in Mathematics Education (CERME 14), Bozen-Bolzano, Italy.

Postelnicu, F. & Postelnicu, V. (2020, Spring). The Camel Problem: An example of computational thinking. *OnCore, Journal of Arizona Association of Teachers of Mathematics*, 82-88. https://www.researchgate.net/publication/342301110_The_Camel_Problem_An_Example_of_Computational_Thinking

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., and Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Sciences Classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147. https://doi.org/10.1007/s10956-015-9581-5

Wing, J. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33–35. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *366*(1881), 3717–3725. https://doi.org/10.1098/rsta.2008.0118