Students' Use of Programming as a Problem Solving Strategy in Probability

Karoline Smucker
The Ohio State University

In this work we examined the effects of programming as a problem solving heuristic on students' mathematical work on tasks involving probability and expected value. Analysis of student performance on a unit of instruction that focused on students' competence in both programming and calculating probabilities and expected values revealed that students can see programming as a valid problem solving strategy and use it effectively.

Keywords: Computational thinking, Curriculum, Probability, Programming, Simulations

Computational thinking is a fairly recent development in the mathematics education research arena. Its origins are in computer science, but Wing (2006) states that it is a "fundamental skill for everyone, not just computer scientists" (p. 33). It includes practices with data, modeling and simulation, computational problem solving, and systems thinking (Weintrop et al., 2016). Drawing on this perspective, this study focused on the utility of programming as a problem solving strategy when examining probability tasks in one mathematics classroom. One broad question guided data collection and analysis: does the use of programming in addition to virtual/physical simulations and theory enhance student understanding of these topics?

Methodology

A unit of instruction was developed and implemented in a precalculus class consisting of 10 students. The curriculum utilized several strategies, including programming, to teach concepts related to probability and expected value. The unit, which consisted of five lessons, first introduced students to several programming concepts, including conditionals and loops. It then included simulations related to three specific games of chance. Upon completion of the unit, students were asked to evaluate the curriculum and what they seemingly gained from the unit.

Results

Students were engaged throughout the unit and showed persistence in solving tasks, which involved modeling games of chance. Post unit assessment results indicated that a majority of students became competent in writing basic programs to solve probability tasks. Students also identified the potential of programming as a complement to other strategies with which they had familiarity. Several students used the word "shortcut" when evaluating programming as a strategy. It appeared that they questioned the theoretical value of programming for solving problems, even if the method yielded the same result. Students used creative strategies when writing programs, some of which were unanticipated by the teacher/researcher.

Implications

Because the sample group for the curriculum was small and the scope of curriculum fairly limited, more research is needed to further expand students' perceptions of and facility with programming when solving problems. With increased interest in inclusion of computational reasoning skills in the curriculum more systemic research is valuable in defining effective ways to help students not only acquire programming skills but also realize its value for extended mathematical work.

References

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.

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