Computational Thinking Tools as a Platform to Teach Mathematics

Marja Miller, Masters of Arts; Dr. Immaculate Namukasa
Faculty of Education, University of Western Ontario

ABSTRACT

As society becomes more technology driven, computational thinking skills are essential in preparing students to be creative, innovative and to develop novel solutions. This project is aimed to answer the research question: What might computational thinking integrated with mathematics thinking look like? The researcher worked with eight K-8 class teachers in five schools implementing CT and Math activities in the classroom. The researcher collected surveys, audio recordings, interviews, field notes and photo data. First year findings show that students are enthusiastic and engaged in the coding and math activities. Students enjoy programming robots, making the characters move in apps as well as changing movement patterns. Use of CT tools in teaching mathematics appears to create more conducive learning environments where students, for instance, are willing to explore, tinker, make mistakes, and troubleshoot.

INTRODUCTION

CT tools are objects-to-think with (Papert, 1980). CT/Math activities can be considered “warm” math tasks because they provide “non-traditional ways of teaching math with variable entry and non-standard rich learning tasks that go beyond more practice and procedure” (Namukasa, Gadannedis, & Cordy, 2009, p.49).

The researcher worked with eight K-8 classroom teachers in five schools and directly collaborated with the teachers to create activities that met the curriculum standards. Collected field notes, teacher surveys, audio recordings of interviews with teachers as well as photos of students’ work and teachers’ presentations.

Research questions:
1. What might computational thinking integrated with mathematics thinking look like in K-8 classrooms?
2. What are its affordances, enablers, and constraints?
3. What are the students’ attitudes towards it?

CT Tool categories explored in the context of Mathematics:
- Block-based/virtual programming languages: digital making, programmable materials and tangible; programming blocks; and programming learning apps, games and simulations.

Students were provided with brief introductory step-by-step instructions on how to engage with the CT tool. They used different skill sets to plan, code, simulate, manipulate, and then challenge themselves or each other at distinct exploration centers, each based on a tool.

RESULTS

Math Concepts Embedded in CT tools: Through CT and mathematics activities students had the opportunity to be creative, innovative and develop novel solutions to mathematics problems.

CT and Mathematics Complementary Concepts:
- Instructional/Steps/code algorithm
- Motion Commands
- Turns, angle measures, direction, magnitude
- Distance, time, speed measurements
- Coordinates planes, grid, orientation, mapping, geometry
- Translations, dilations
- Clock direction and measures
- Iterations, loops
- Debugging/checking for errors
- Problem solving
- Logic, Flow charts
- Functions and Variables
- Expressions and algebra
- Basic math operations, order of operations
- Conditional statements: If, then, else

Student Participants:
- Eager to touch the materials, although reluctant to wait for or read instructions. Were willing to listen to video based manuals.
- Eager to remotely drive robots, to code, to enact code, and to set challenges with CT tool.
- Okay with making mistakes, able to troubleshoot and problem solve, frustrated when technology did not work.
- Tinkered when playing with robots, created math games/apps, and imagined other possibilities.
- Engaged in multiple ways with the CT tools—some comfortable with enacting code developed by others.
- Multiple sessions, more time to explore/discover reinforced the CT/Math concepts taught, and made them fluent by the third center rotation as they took less time and asked fewer questions.
- Made connections between the programming language used in one tool in Scratch to that used in a physical programming kit such as Kibo block programmed Robots, a sign of abstracting concepts.
- Centers also provided more for the students to explore as well as communicate (save, write and talk).

Teacher Collaborators/Participants:
- Used own teaching practices
- Engaged with the students, challenged students to try new tasks in small groups
- Observed the activities and gave feedback on future lessons that could be taught (e.g. using the Cuberto world map to not only teach programming, but getting students to think about coordinate planes and connect that to story telling and social studies).

Figure 1. A chart with vocabulary for a Grade 2 and 3 math lesson

Figure 2. Navigating a maze with hybrid user interfaces of physical programming OSMO blocks, accessories and apps

Figure 3. A grade 8 student writes code for calculating surface area in Scratch

REFERENCES


CONCLUSION

There is potential for CT tools to support the learning of mathematics curriculum among other subject areas as well as to excite students about learning concepts.

It helps when sufficient time is explored to be given to students. This in turn reinforces the teaching of mathematical and CT concepts.