JOURNAL OF RESEARCH IN SCIENCE, MATHEMATICS AND TECHNOLOGY EDUCATION
DOI: 10.31756/jrsmte.213SI

STEAM Camp: Teaching Middle School Students Mathematics, Science and Coding through Digital Designs

This paper was awarded third place in the second special issue of JRSMTE on graduate research

Marja Gabrielle Bertrand, Immaculate Kizito Namukasa & Li Li

Western University, Canada

Abstract: In this study, we explore how to teach mathematics, science and coding through digital tools, design projects, and global competencies. We explore the question: How do upper elementary school children develop an understanding of mathematics and science coupled with coding through digital design? The theoretical framework adopted for this study is Kafai and Burke's (2014) definition of Computational Participation: a shift from code to actual applications; a shift from tools to communities; a shift from starting from scratch to remixing; and a shift from screens to tangibles. We conducted a qualitative case study interlinked with Design-Based Research. Both STEAM camps were an outreach program for students in grades 4-8 in Ontario, Canada. The two camps were designed and facilitated by a research team from the Faculty of Education. The research team developed the curriculum through an iterative process (design-test-revise-repeat). There were 43 students registered in the STEAM camps, and 34 of them participated in the study. We collected observation, interviews, audio/video recordings, and survey data as well as pictures of the students' work. Our main findings were that students were provided with opportunities to: 1) develop a deeper understanding of curricular concepts; 2) engage more with the digital tools when they were remixing, improving, and reimaging the design; and 3) apply their knowledge to global competencies. The findings of this research have implications for improvements in researching, designing, and implementing design projects as part of a pedagogical approach to teaching mathematics and science, coupled with coding, in an interdisciplinary context.

Keywords: *Mathematics education; Science education; Coding; Computational Participation; Global competencies; STEAM education.*

Introduction

In this study, we explore how to teach mathematics and science, coupled with coding, through digital tools and design projects: specifically, using coding as an alternative way for students to express and collaboratively apply their mathematical thinking and understanding. We also examine how students can develop a deeper understanding of mathematics, science and technology through global competencies, specifically the United Nations (UN) sustainable development goals (https://ncte.org/blog/2019/08/establishing-a-framework-with-global-competencies/). The findings reported have implications for improvements in researching, designing, and implementing design projects as a pedagogical approach to teaching and for optimizing learning in both in- and out-of-school contexts. This paper addresses the following research question: How do upper elementary school children develop an understanding of mathematics and science, coupled with coding, through the digital design of real-world problems and contexts?

Coding and mathematics

Many educational scholars have talked about how coding has been integrated into K-12 mathematics education (e.g., Colgan, 2020; Goldenberg & Carter, 2021; Higginson, 2017; Israel & Lash, 2020; Miller, 2019). Diverse opinions have emerged concerning the nature of the learning that coding can engender in mathematics. Many researchers argue that coding has a positive effect on mathematics learning (e.g., Barichello, 2016; Rodríguez-Martínez et al., 2020; Sung et al., 2017). For example, Barichello (2016) found that coding offers students opportunities to engage in more "genuine mathematics" through problem-solving (p. 796). OME (2020) states that "coding can be incorporated across all strands and provides students with opportunities to apply and extend their mathematical thinking, reasoning, and communicating" (p. 85).

In contrast, some scholars cite the challenges of learning coding in a mathematics classroom (Dohn, 2020; Ke, 2014; Resnick & Rusk, 2020). Dohn's (2020) research, for example, found that specific coding tasks with controlled structures that limit students' autonomy "can have a negative impact on students' interest" (p. 81). Moreover, according to Ke's (2014) research, students' interactions with the curriculum content may be interrupted when storytelling or game-world crafting predominates students' attention (i.e., students focus more on the coding technology than on the mathematics concepts). To avoid this pitfall, Goldenberg and Carter (2021) suggest that educators "let mathematics, not coding, drive the design, with coding and mathematics [being] mutually supportive" (p. 981). According to Resnick and Rusk (2020), "in many educational settings, coding is introduced in much more limited and constrained [ways], so that students do not have the opportunity to experience the full conceptual and expressive powers of coding" (p. 121). For example, sometimes coding is introduced to students by getting them to just copy the code, rather than allowing them to explore, modify, experiment, remix, and debug the code (Resnick & Rusk, 2020). Resnick and Rusk (2020) suggest that for students to develop a deeper understanding of coding they should "create projects, based on their passions, in collaboration with peers, in a playful spirit" (p. 120). Further, Goldenberg and Carter (2021) explain how "programming can be an advantageous third language [i.e., alternative way to communicate outside of written and oral language skills] to help restore mathematical connections" (p. 969). Similarly, diSessa (2018) described "computation [through coding and computer programming] as a new literacy" (p. 1) or a language for students to express their ideas and mathematical thinking.

Global competencies in K-12

According to Cotton (2001), "mathematics is a powerful tool in explaining and interpreting the world in which we live . . . [and] is an important tool or 'language' in constructing, explaining, and interpreting the globalized world" (as cited in Schell-Straub, 2013, p. 10). The Ontario Mathematics, Science and Technology curricula, K-12, has a specific section for environmental, human rights, equity, and inclusive education to address global competencies and sustainable development goals. One of UNESCO's (United Nations Educational, Scientific and Cultural Organization) main goals is to "ensure that all learners acquire the knowledge and skills needed to promote sustainable development . . . through education for sustainable development [goals] and lifestyles" (p. 287). In this study, we examined how students can learn about global competencies and sustainable development goals in the context of mathematics and science education. We also investigated how coding can be used as a tool or vehicle to mathematically model or solve a real-world problem.

Context of the study

Our study took place in Ontario, where the government introduced a new mathematics curriculum that integrates coding and computer programming for Grades one to eight in 2020; in 2021 it was integrated into the high school curriculum (OME, 2020; 2021). This curriculum reform was extended to the science and technology curricula in 2022 (OME, 2022). Ontario was not the only province in Canada to integrate coding into the curricula (mathematics, science, and technology): Nova Scotia (NSDE, 2015) integrated it into the information communication technology and mathematics curricula and British Columbia (BCME, 2016) incorporated it into their design arts curriculum. At the time of our study, Nova Scotia (NSDE, 2015), British Columbia (BCME, 2016), and Ontario (OME, 2020, 2021, 2022) had implemented coding and maker education curricula in elementary schools. Further, similar to other countries globally, the Council of Ministers of Education in Canada was taking initiatives to incorporate global citizenship and sustainable development goals into the elementary and high school curricula (CMEC, 2020).

Theoretical frameworks

Our work is grounded in the view of learning as a constructionist experience. Papert and Harel's (1991) constructionism theory of learning is foundational to maker education (i.e., students learn while making) and guides us in theorizing about students' learning of functional and disciplinary knowledge when using computing tools. Papert's work has been further developed to focus on different aspects of thinking with computers, including computational design for personal expression (Resnick, 1998), computational literacy (diSessa, 2000, 2018), and computational making for improved participation in disciplinary and real-life practices (Kafai, 2016).

The theoretical framework adopted for this study is Kafai and Burke's (2014) Computational Participation (CP). We used this as an analytical lens to study the different methods used to engage students with digital tools to express their mathematics/coding knowledge and understanding. Kafai and Burke talk about the transition from Computational Thinking (CT) to CP across four dimensions (Burke et al., 2016, p. 374): 1) a "shift from code to actual applications" through the creation of design projects; 2) a "shift from tools to communities", such as a collaborative community of their peers that share resources and ideas around these virtual spaces; 3) a "shift from starting from scratch to remixing" where students improve and reimagine the design; and 4) a "shift from screens to tangibles" in which students code and program tangibles such as microcontrollers and sensors. We also draw from the Ontario Mathematics, Science and Technology curriculum (OME, 2007, 2020, 2021, 2022) and UNESCO's (2016) Sustainable Development Goals to examine the specific disciplinary, interdisciplinary, and transferable skills learned in the STEAM (Science, Technology, Engineering, Arts, and Mathematics) camps.

Methods

Research design

We conducted a qualitative case study interlinked with Design-Based Research (DBR). For DBR, we used an iterative design cycle when creating and implementing a curriculum by trying it out and refining the design based on the instructor and student feedback (Cobb et al., 2003). Since the first STEAM camp in 2021 which was conducted for online synchronous learning (due to the Covid 19 pandemic), there have been further iterations of the curriculum implemented in 2022 that were designed for in-person learning. Both STEAM camps were outreach programs for fourth to eighth grade students in an out-of-school context.

The curricula for the camps were designed and facilitated by the research team composed of faculty, graduate and undergraduate students from the University's Faculty of Education in Ontario, Canada. The planning for the camps involved the design of the modules (i.e., presentations with hyperlinks to video illustrations, themes, physical and digital materials, and website apps), the inclusion of a physical STEAM kit (only for Year 1) and educational resources assembled by the research team, which participants picked up from a location at the university. The STEAM kit included a tangible coding micro-controller − the micro:bit (https://microbit.org/), custom mathematics pages, graph paper, crafting materials such as playdough and a bingo dabber, and hands-on STEM/STEAM/Maker education tools. Overall, curriculum design was based on Papert and Harel's (1991) constructionism theory of learning in which students learn by doing and constructing their own knowledge when making or creating a product of interest.

Participants

The invitations to the two STEAM camps were posted on the following platforms: Twitter, Facebook, an electronic newsletter for students and staff at the university hosting the camps, and a Google website created by the research team. To protect the participants' anonymity, the students were labelled with codes and the year of the study (e.g., P01, Year 1).

In the first year, 2021, two instructors created the curriculum and facilitated the lessons and three volunteers kept track of the chat box and answered students' questions. A total of 22 students were registered for the camp, from Ontario and British Columbia (two Canadian provinces). There were zero Grade 4, nine Grade 5, seven Grade 6, four Grade 7, and two Grade 8 students. Of those registrants, eighteen students consented to participate in the study. Of those participants 55% were female and 45% were male. Most students (17 of the 18 participants) had some experience with coding in school (e.g., Scratch coding software) but not with the specific digital tools in the camp (Micro:bit, Tinkercad and Cospaces Edu).

During the second year, 2022, two instructors created the curriculum and facilitated the lessons and two volunteers handed out materials, answered students' questions, conducted informal interviews, and took photos of student work/projects. A total of 21 students from Ontario, Canada registered for the camp. There were seven Grade 4, five Grade 5, four Grade 6, zero Grade 7, and five Grade 8 students. Of those registrants, 16 students consented to participate in the study. Of those participants 29% were female and 71% were male. Since the camp was in-person, the researchers were able to informally determine participants level of comfort with coding and the programming software through conversations and observations. Out of 21, there were seven beginners, twelve intermediate, and two advanced level coders.

Table 1

The learning environment and curriculum development for the two STEAM camps

Program	Learning Environment and Curriculum
Overall	The two STEAM camps were an outreach program for students in grades four to eight in an out-of-school context. The purpose was to teach mathematics coupled with coding in an informal learning environment by using physical and virtual manipulatives, and dynamic mathematics software. We utilized a variety of virtual materials and manipulatives (e.g., Scratch coding software and Cospaces Edu making and design software), website learning applications/software – learning math apps from websites and dynamic software, online collaborating boards and polls, and computational thinking tools (http://cscircles.cemc.uwaterloo.ca/console/).
Year 1	The first STEAM camp was delivered virtually using a web conferencing platform, Zoom (www.zoom.us), for a total of three days (i.e., one module per day) in August 2021, with synchronous meetings lasting 2 hours for each module. Each day of the camp focused on a particular digital tool (e.g., Day 1: Micro:bit, Day 2: Tinkercad, and Day 3: Cospaces Edu). The research team integrated coding, computational thinking, and global competencies into the curriculum design of the lessons/activities.
Year 2	The second STEAM camp was delivered in person from 9 am - 12 pm Monday to Friday for one week in August 2022. The camp activities took place in the Art room at the University's Faculty of Education. The room was spacious, larger than a regular sized classroom (can seat 30-40 students), with round tables, mobile chairs and whiteboards (with wheels). The room had lots of tables and floor space for the students to move and interact with digital tools and physical materials. For the onsite camp, we utilized a variety of additional materials such as physical manipulatives (including, craft materials, Merge Cube and VR goggles), and spherical robots (Sphero and Ozobots). Some of the materials such as the robots and the micro:bit were hybrid tangible materials (i.e., both physical and virtual).

Learning environment and digital tools

The activities in the five modules taught a variety of tools (see Table 2) such as physical manipulatives, virtual manipulatives, and dynamic mathematics software. We designed the modules to include in-depth mathematical concepts and cross-curricular connections, involving coding, design, and selected STEAM concepts. Further, we incorporated some of the UN's Sustainable Development Goals (https://sdgs.un.org/goals) in the design challenges to provide students with a real-world context, such as working toward developing safe, sustainable housing, inclusive communities (people who are underrepresented or minoritized feel welcome, safe, and a part of their community), and renewable energy. The curricular concepts were embedded into each activity to "let mathematics [and science], not coding, drive the design" (Goldenberg & Carter, 2021, p. 981) in the modules.

The research team intentionally created activities with the broader goal of promoting equity, diversity, and inclusivity and improving the participation of *all* learners (Kafai, 1996). We created activities that encouraged students to explore mathematics in different cultural artefacts (e.g., tessellations in Islamic, African and Indigenous art and architecture), share stories about their culture and background, and integrate these explorations and stories into their digital designs through symbols and images which they selected based on their meaning to them. We "developed activities that integrate mathematics and coding in the classroom to enhance the student's overall learning experience and make it more meaningful" (Bertrand, 2019, p. 28). In the STEAM camps, each digital tool was intentionally selected to support "low floor, high ceiling, wide walls" activities to allow entry for all students of varied backgrounds, grades, and comfort levels to learn mathematics and coding (Gadanidis, 2015). Each activity had simple ('low floor') and complex ('high ceiling') tasks and multiple ways to approach a problem ('wide walls'). One instructor reflected, "It's really low-floor, high-ceiling based activities [in the STEAM camps that provide] . . . a lot of entry points for each and every student" (Instructor 1, Year 1).

Table 2

Details on the specific digital tools and apps used for both STEAM Camps

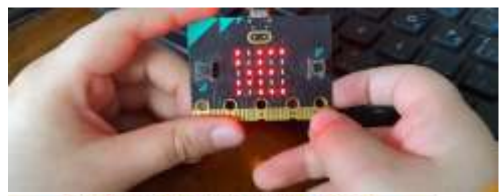
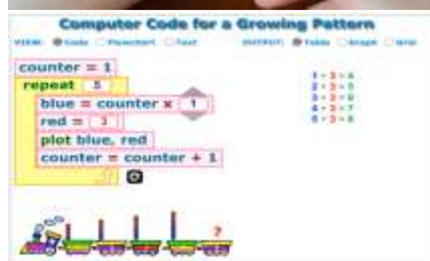
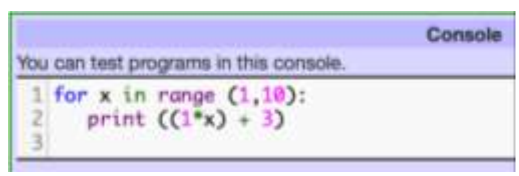
	Digital tools and apps	Sample image
Year 1		
	Micro:bit: a pocket-sized computer with built-in sensors (e.g., Bluetooth, light, temperature, compass, radio and accelerometer) and can be programmed using the free MakeCode software at https://microbit.org/ or http://makecode.com	
	Math and Coding Apps on Patterns and symmetry: https://imaginethis.ca/ or https://researchideas.ca/mathncode/sims-growpatt.html	
	Python and Text-based consoles: http://cscircles.cemc.uwaterloo.ca/console/ or https://replit.com	
Years 1 & 2	*Tinkercad:* an open-source online software for children to create 3D models by combining solid geometric shapes in order to make more complex figures. The files can be downloaded for 3D-Printing or Cutting at https://www.tinkercad.com/	

Table 2 continued

Years 1 & 2	*Cospaces Edu:* an online app that provides children with the opportunity to "build their own 3D creations, animate them with code and explore them in Virtual or Augmented reality" https://cospaces.io/edu/	
Year 2	*MERGE cubes, VR goggles and AR books:* Students can apply their understanding of 2D and 3D geometric shapes, measurement, estimation, scale factor, angles, rotation, and translations as they play with the virtual image projected on the MERGE Cube https://mergeedu.com/cube	
Year 2	*Scratch:* a "high-level block-based visual programming language and website aimed primarily at children as an educational tool, with a target audience of ages 8 to 16" https://scratch.mit.edu	

Data collection and analysis

The research team collected the following data: surveys (i.e., daily reflections in the Near pod- https://nearpod.com/, and Google forms), observations (using an observation template), interviews with students and instructors (transcribed verbatim), and pictures of students' work and projects. We transcribed 25 audio-recorded interviews: each transcript ranged from 5 - 45 minutes (student interviews were done during the camp and were much shorter than the instructor interviews). We analyzed a total of 272 survey questions, eight curriculum documents for a total of 553 power point slides, six observation documents for a total of 26 pages, and 509 photos of the students' work and projects. *Organizing and Preparing Data:* The photos were organized based on the day and the digital tool (e.g., Day 5 AR & VR slides), coding levels (e.g., simple or complex code), and preexisting themes (e.g., Heritage Culture and Storytelling). The surveys, observations, and interview data were organized based on when the data were collected (Years 1 or 2) and the source of the data (student or instructor). The audio and video recordings were transcribed using Otter.ai (https://otter.ai/) and then checked by hand for any discrepancies.

Coding the Data: The data were coded using Nvivo software (https://lumivero.com/products/nvivo/) for the surveys, observations and interviews. For the initial coding of the data, we used pre-existing themes from the literature and theoretical frameworks (Hubert, 2014). During the second round of coding, we analyzed the data using new themes and codes that emerged from the data (Hubert, 2014; Parker et al., 2017), such as the theme "remixing, improving and reimagining the design." After the second round of coding, relationships were identified among the codes resulting in categories and subcategories (Parker et al., 2017), such as concrete applications (category) being further refined to mathematics, coding and programmable tangibles (subcategories). The process can be described as an iterative and inductive coding procedure using a constant comparative method which involves re-coding the data to further refine the themes (Glaser & Strauss, 2017; Strauss & Corbin, 1997). *Triangulation of Data:* We analyzed the data simultaneously from different sources, such as written surveys, observational notes and reflections by research team members, transcripts of video/audio recorded interviews, and pictures of students' individual and group work and projects. To increase the reliability of the data analysis, the primary investigator on the research team cross-checked the codes with other data sources (Gibbs, 2007; Guest et al., 2011).

Results

We analyzed how students learn in the context of mathematics and science education coupled with coding when students designed and created projects with real-world problems and global competencies. Our main results are presented under four topics: Knowledge, communities, re-imagining designs, and tangibles.

Applying knowledge in the design projects

Students learned computer programming skills when they created and wrote code to animate their creations in Cospaces Edu. They were also able to learn transferable skills when applying knowledge from one lesson or coding software to another. One student said, "I learned that you can code Spheros like Scratch" (P26, Year 2). One instructor described the student learning experience as "something meaningful, that [they] have to apply mathematics in these very playful and very related [ways to] . . . their own life" (Instructor 3, Year 2). Students shared their ideas about: "What math is involved when designing and programming animated characters in video games?" (Cospaces module, curriculum document). Students talked about how angles and their functions, speed, distance, curve trajectories, and basic mathematics operations are involved in coding the animated character to jump, run, flip, move forward, etcetera. Specifically, students learned how to animate their objects: change direction by moving left, right, up, or down, clockwise, counter-clockwise, change the speed by moving camera x distance in y seconds, rotate around the x, y and z-axes (see Figure 1).

One instructor noted, "When we use the UN Sustainable Development Goals as a catalyst for learning, it really puts empathy at the focus, which is right at the root of both design thinking and engineering design process" (Instructor 1, Year 1). For example, when students were asked to create a sustainable home in Tinkercad, they had to consider environmental factors, the local community, and the people they were designing the house for. The design challenges allowed students to apply the digital tools learned to design software solutions to real-world problems, by using mathematical and scientific concepts (see Figure 2). For example, students needed to create an entry-exit to their home (e.g., create a doorway with specific dimensions, such as length and width, that meet the Ontario building code standards); create sustainable housing (e.g., solar panels on the roof for renewable energy or include skylights/large windows with a tall structure for vertical food and herb gardens in the home); use three different shapes (e.g., choose different 3D shapes to design an aesthetically pleasing home, use spatial reasoning and choose dimensions that would maximize the volume of the house); and be able to group these shapes as seen in the design criteria (e.g., students learn skills such as grouping geometrical shapes in Tinkercad that can be applied to other design software). Students also had the opportunity to make cross-disciplinary connections where they used elements and principles of design (arts), 3D modelling using geometrical shapes and transformations (mathematics), and coding to animate the 3D objects or to run number patterns (digital literacy skills).

Figure 1

One student used repetition (a forever loop) to code two raccoons to move at 4.5 metres per second (speed) and rotate 90° clockwise around the x, y, and z axes.

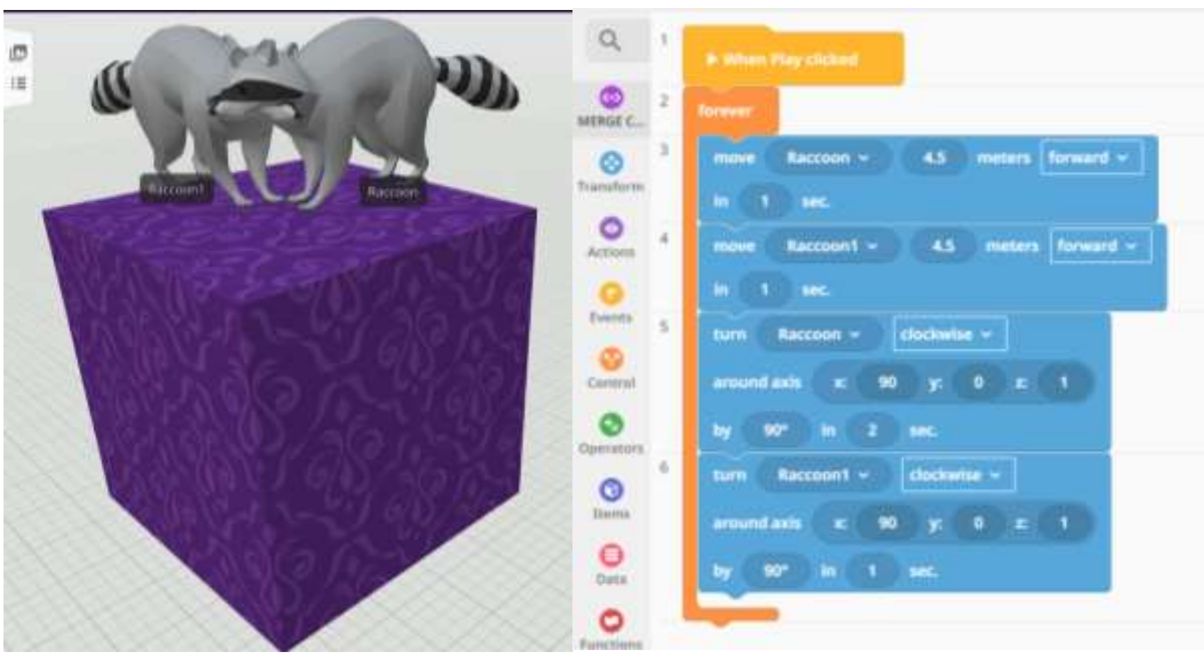


Figure 2

Screenshot of presentation slide: UN Sustainable Development goals were embedded into the design projects.

Sharing resources and collaborative communities

Students shared what they made and how they did it. They appeared to be eager to help other students when they were having difficulties. One instructor stated, "Overall, they were so eager to share their ideas . . . They were motivated to explore many features by trying, asking questions, and sharing ideas with each other" (Instructor 3, Year 2). Another instructor reflected, "I remember from the camps . . . [they were] reluctant [at first to share], and then everyone wanted to share their design, yes, look what I did [and] I added this feature . . . And that is so powerful" (Instructor 1, Year 1). An instructor asked "Can we see your code how it works? If you could just press play, then we'll be able to see your animation" (Instructor 2, Year 1). One student designed and recreated his favorite 'stuffie' (stuffed toy) called a Dogeburger using 3D geometric shapes, scaling (stretching the image vertically to create the Dogeburger skyscraper) and merging different geometric shapes via the group feature in Tinkercad (Figure 3). He shared his screen with his peers and explained step-by-step how he imported the Dogeburger skyscraper from Tinkercad into Cospaces Edu (Figure 4). He was very excited to share with his peers what he had made in Tinkercad and Cospaces and to share how he created it (e.g., the grouping of two shapes or cutting a hole in a shape, and scaling of the objects).

Figure 3

A student created a Dogeburger skyscraper using 3D geometric shapes, scaling, and merging shapes together.

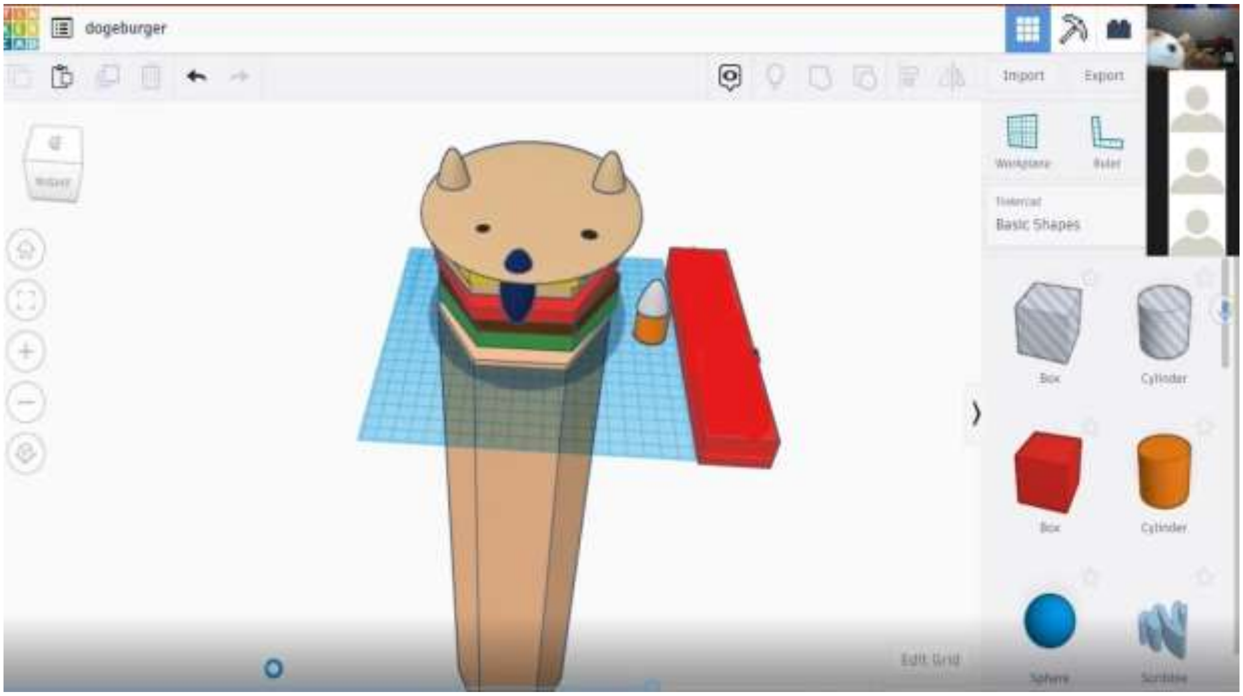


Figure 4

The student imported the 3D object (Dogeburger) from Tinkercad into Cospaces Edu and duplicated/rescaled it.

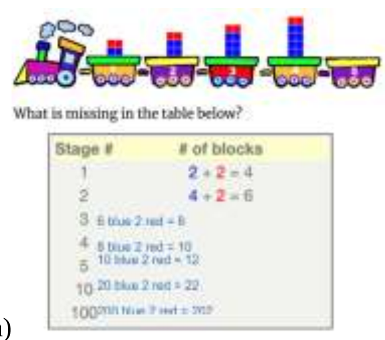
Remixing, improving and reimagining the design

In the Micro:bit module in Year 1 (2021), students learned about patterns and relations and used mathematical modeling with a text-based coding mathematics app to represent the pattern (see Figure 5). Students represented their understanding of patterns visually using numbers and graphs and then were guided to code the concepts. As a class, students discussed how patterns occur naturally in nature and other real-world examples (integration of mathematics and science curricular concepts). One instructor explained, “So we had students construct patterns, [such as] . . . coding like a loop pattern in Python setting like, what a table of values looks like, even just by printing like the levels one to ten, and then getting them to do basic operations and math” (Instructor 1, Year 1). Students had the opportunity to explore two ways of representing the code for the pattern of 2, 4, 6, 8, 10 . . . (even numbers) as seen in Figure 5 (b) and (c). Method one could be seen as a more efficient way to represent the pattern through multiplication rather than addition. Students had the opportunity to represent these patterns using arithmetic series where 2 is added to the previous number in the sequence each time. Students were also able to represent this pattern as a geometric series where the term number is doubled or multiplied by 2 to get the next number in the sequence. One student simply stated that “the patterns related to math and also the coding was very fun” (P10, Year 1). Some students talked about the specific mathematical concepts that they learned: “I learned some different types of patterns that use addition and multiplication” (P01, Year 1). Similarly, another student expressed “I learned that [patterns are] related to math and learning [mathematics] was [related to] the Python coding [i.e., text-based coding software], of going up by 5 or 10. I felt a bit frustrated in the beginning, but then it [Python] turned fun when I learned how to use it” (P14, Year 1). “So instead of just coding for the sake of coding, it's coding with a defined purpose” (Instructor 1, Year 1).

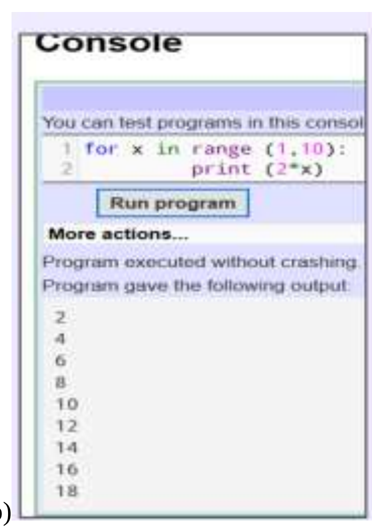
Figure 5

Students represented the same pattern a) numerically and using text-based coding in Python through b) multiplication, and c) addition.

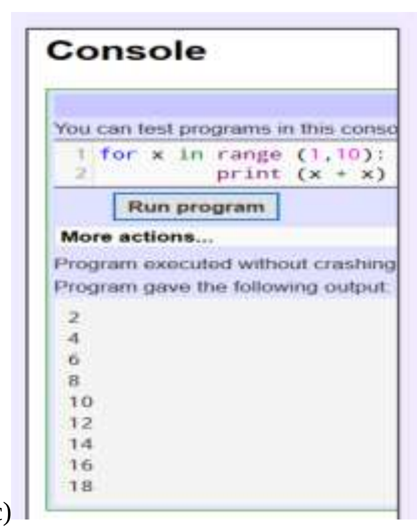
a)



b)



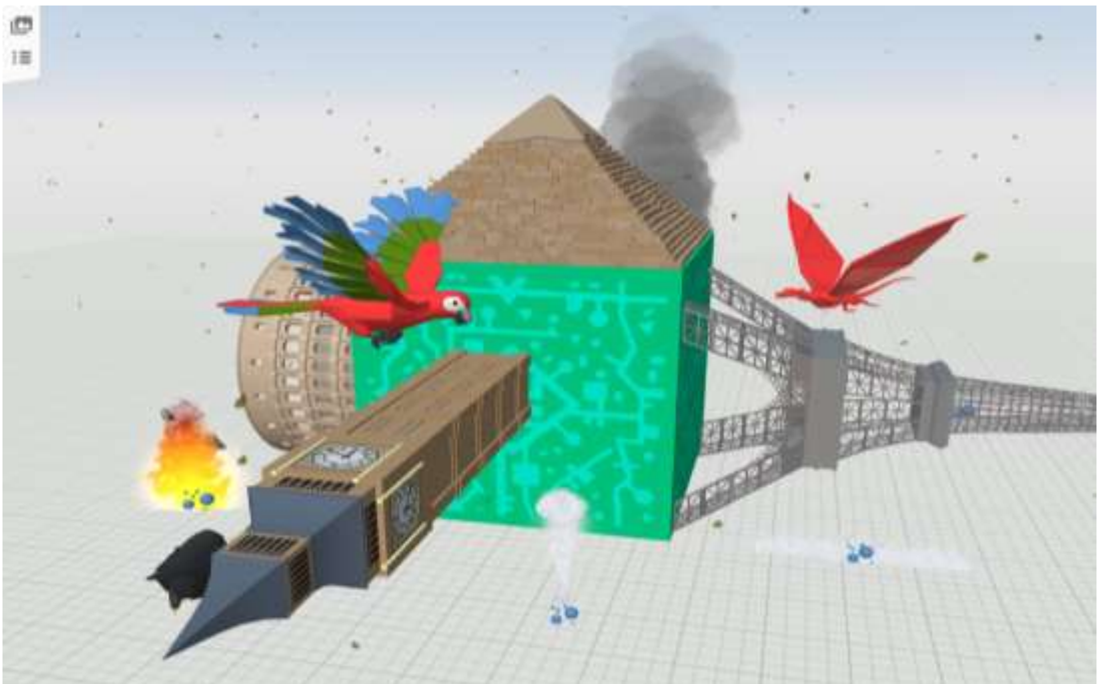
c)



In the second STEAM camp (2022), students reimagined their design 3-dimensionally and created their own augmented reality by superimposing the image onto the MERGE cube. One student shared, “I learned how to animate stuff in Cospaces. I could do my own MERGE cube [and AR/VR technology] … I like doing Cospaces, because it allows you to create virtual worlds and stuff” (P26, Year 2), as seen in Figure 6.

Figure 6

Students incorporated geometric cultural monuments, landmarks, and animated characters into the AR designs in Cospaces (e.g., Big Ben in London England, the Eiffel Tower in Paris France, and the Sphinx of Giza in Egypt).



Mathematics, coding and programmable tangibles

One instructor explained “I think it's a really good mix of concrete applications and math tools, like manipulatives, and then also representations in a digital space” when students are coding and programming (Instructor 1, Year 1). Students also made the connection between coding and mathematics. When the students were asked “What did you learn that is related to math, coding, and other learning?” (Survey question), they replied that “math is coding” (P18, Year 1), you “need to make calculations to know where you want your object” (P14, Year 1) to go, and that “the way to make code . . . is basically related to math” (P10, Year 1). Another student described his experience: “I thought the micro:bits were fun to use. The whole class was basically a math class because coding is just all numbers and numbers are math” (P08, Year 1). One student saw a relation between the math learned and the “timing and calculating the speed” when coding the robot (P19, Year 2). Similarly, another student expressed that “the math that was used in coding [the Sphero robot] were angles, timing and movement” (P24, Year 2).

After the sessions were done, several students continued to explore the apps, such as with the Micro:bit, Tinkercad, and Cospaces during their evenings between the camp days. Students also took the initiative to look in greater detail at the slides in the modules and to watch other online videos that were hyperlinked in the module presentations or those found on the websites for the apps. As a result, student learning appeared to extend beyond the context of the STEAM camps. For example, one student made a digital game with the micro:bit, stating that the session "helped [him] learn micro:bit coding and . . . make [a] space invaders game on [the] micro:bit" (P18, Year 1) as seen in Figures 7 and 8. This student used mathematical concepts like number patterns, timing, operations, angles, and movements, explored in earlier modules. During the creation of the game, he used more advanced mathematics concepts such as x- and y- coordinates, and conditional statements of the "if this" condition (equal to or greater than) then "do" (change score by 1; y = 0) as seen in Figure 7. He was also able to physically play with the micro:bit to try out the code to see if it worked (see Figure 8).

Figure 7

A student explored advanced mathematical skills when he coded the micro:bit and created a space invader game in Make Code programming software.

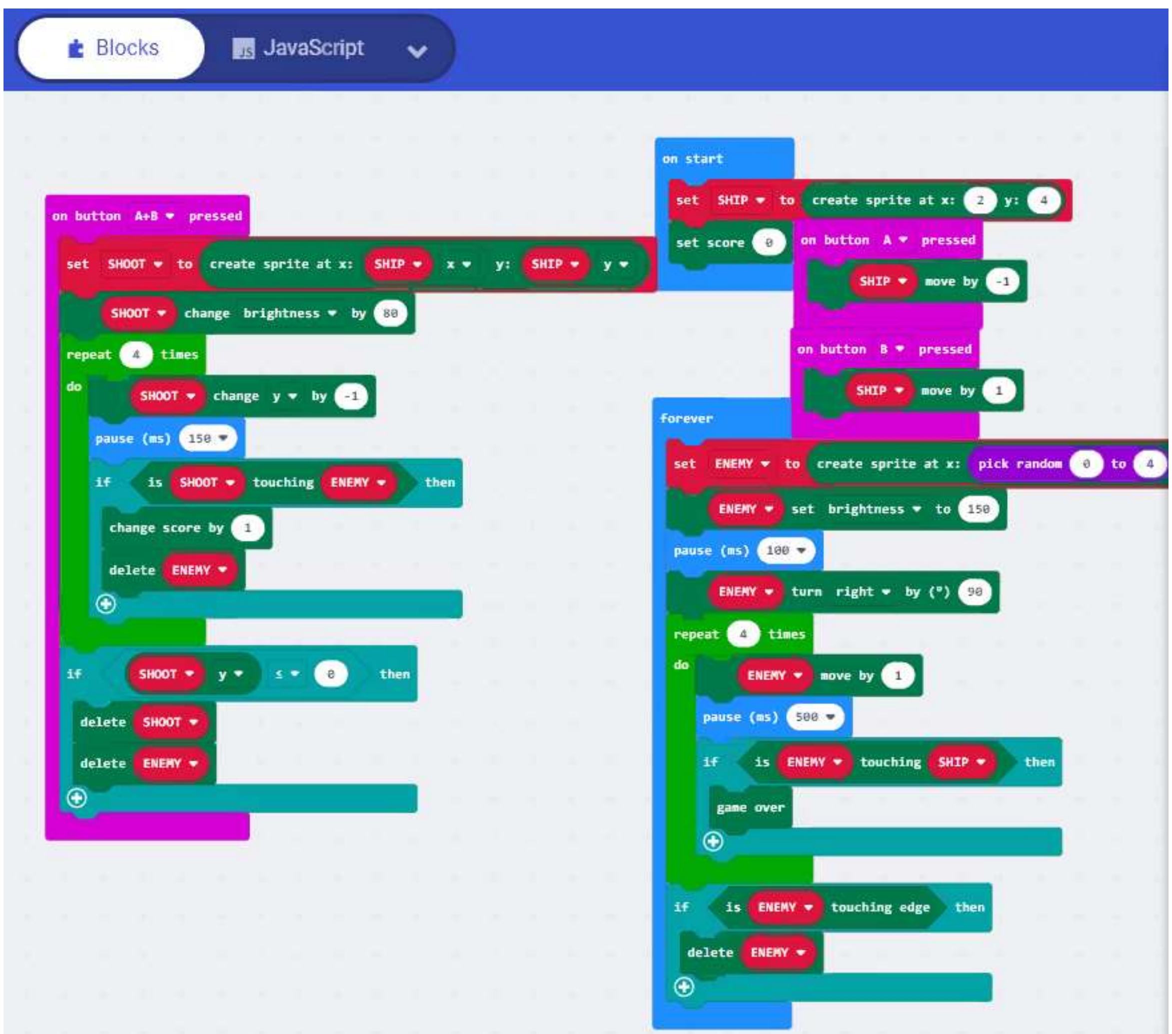
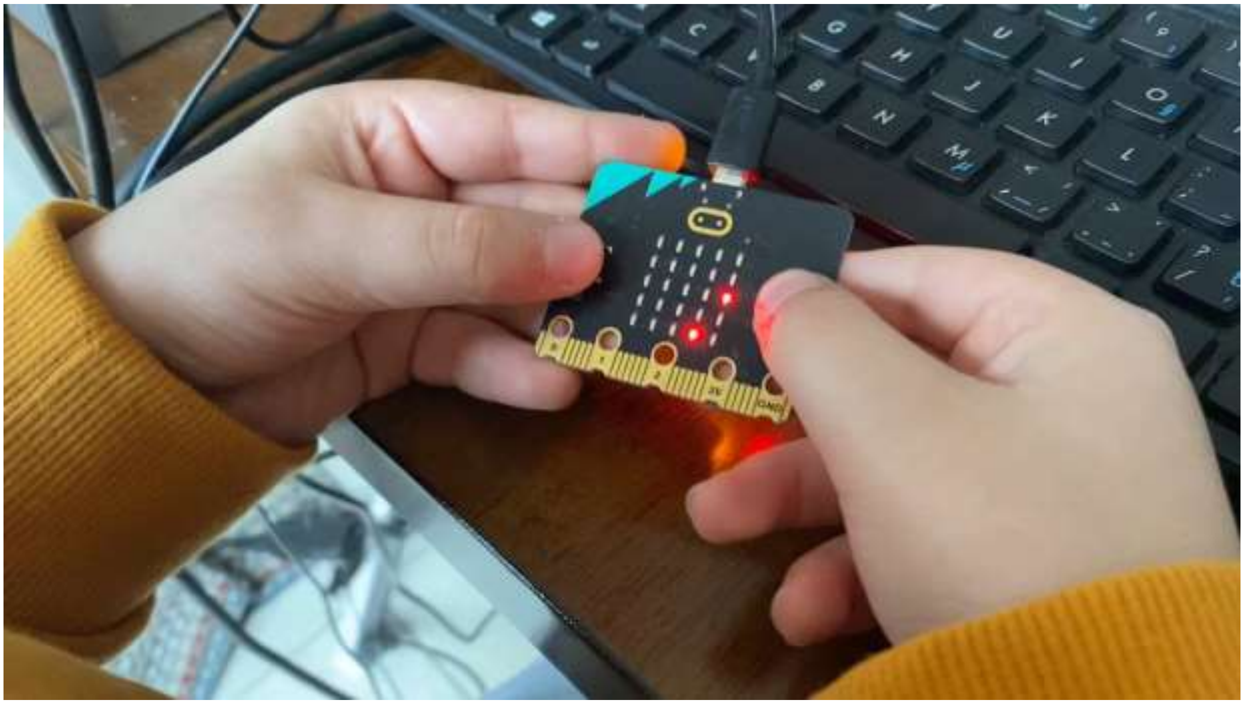


Figure 8

A student (same participant mentioned in Figure 7) was able to play with the micro:bit and learn debugging techniques when the space invader game did not work.



Discussion

In this section, we discuss how upper elementary school students develop a deeper understanding of mathematics, science, and coding through digital design in the results above.

Mathematics through coding and patterns

The students in the STEAM camps appeared to have the "opportunity to experience the full conceptual [i.e., the math, coding, design and other disciplinary concepts] and expressive powers of coding" (p. 121) through their creations, app simulations, and animations (Resnick & Rusk, 2020). This is in line with Barichello's (2016) findings that coding offers students a more "genuine mathematics" learning experience (p. 796). OME (2020) asserts that "students develop algebraic reasoning through working with patterns, variables, expressions, equations, inequalities, coding, and the process of mathematical modelling" (p. 85) when they simulate real-life situations and solve real-world problems. Specifically, in the number patterns activity, students were asked to recognize the patterns (e.g., arithmetic 2, 4, 6, 8, 10 as seen in Figure 5 (b) and (c)), predict what number would come next and compare patterns, and learned to model this pattern using text-based coding (see Figure 5). Also, the curriculum design of the modules used different apps, compared to learning one app in-depth. This appeared to contribute to this focus beyond learning to use coding tools, but rather the application of those tools to STEAM curricula content (e.g., applying skills learned in Scratch to coding with the Sphero robot).

Students explored coding and mathematic concepts such as coding number patterns in Python and conditional "if-then-else" statements in MakeCode software (http://makecode.com). Students also had the time to tinker with, learn by trial and error, learn from sample code, and correct mistakes (debug) in their code if it did not run properly, or produced different patterns than they had intended during the modules. This is in line with Resnick and Rusk's (2020) suggestions for the effective implementation of coding in the classroom by allowing students to explore, modify, experiment, remix, and debug the code. Further, Kafai and Burke (2014) found that getting students to remix and reimagine their designs increased students' computational participation as critical thinkers and producers during their engagement with the materials. These findings also echo Goldenberg and Carter's (2021) idea that "programming can be an advantageous third language to help restore mathematical connections" (p. 969). The researchers "let mathematics, not coding, drive the design [of the modules], with coding and mathematics [being] mutually supportive" (Goldenberg and Carter, 2021, p. 981). This can be seen in the following examples: *patterns and relations* coding activity (Figure 5), *translations, rotations, and scale factor* of the 3D image in Tinkercad (Figure 3) and making connections between *mathematics and gaming* (Figure 7).

Deeper understanding of curricular concepts

In the STEAM camps, students appeared to learn "when and how to use appropriate tools to understand and model real-life situations, predict outcomes, and solve problems" in the context of mathematics education (OME, 2020, p. 103). For example, students appeared to use number patterns, mathematics operations, geometrical transformations of rotations, translations, and scale factor, coordinate geometry of 2D planes and 3D spaces as well as x, y, and z axes (as seen in Figures 1 and 3) and real-world connections (i.e., to simulations, animations, and gaming). Students participated in "low floor, high ceiling, wide walls" activities (Gadanidis, 2015, p. 308). Each activity had multiple entry points with both simple ('low floor' of studying code of simple patterns, designs, and making code involving a few programming concepts, animating, and simulating patterns in the apps) and complex ('high ceiling' of coding and designing simulations and animations with patterns and objects) tasks. For example, some students used premade figures and others designed complex *3D* objects in Tinkercad and imported these objects into their dream place created in Cospaces (Figures 3 and 4). Similarly, some students remixed code and others created a more complex code (Figures 1 and 7). These activities included 'wide walls' of coding simulations and animations in created spaces or games with multiple codes, patterns, and 3D objects. For example, students had multiple ways to approach and represent a problem, such as the design challenge in Figure 2 (different designs, geometric shapes, and renewable energy sources) and the micro:bit activity in Figure 7 (different types of coding languages – Blocks and Java Script).

In general, students' participation in the modules for the STEAM camps straddled between controlled structures that often limit students' autonomy, and less controlled play, which was framed by curriculum expectations, in web-based expressive storytelling and game-world crafting activities (Ke, 2014). The intent of exploring the mathematics and coding apps (see Table 2) was to enable upper elementary school students to actively experience and engage in mathematical thinking and modelling, with both curriculum content and cross-curricular learning. When students moved on to design, code, tell stories and express themselves in the less controlled learning environment, they were encouraged to be creative through the storytelling and game-world crafting activities of the design and coding software. The context of design and programming activities provided an alternative to written and oral language to help students make deeper mathematical connections when they created, simulated, and animated objects in imagined or modelled spaces (Goldenberg & Carter, 2021), as well as to focus their attention on mathematics and coding, and related learning such as making mathematical patterns and digital designs.

Global competencies and sustainable development goals

Students also learned concepts from the Science and Technology curriculum (OME, 2007). One instructor described integrating global sustainable goals into the curriculum as a "catalyst for learning" (Instructor 1, Year 1). In the Tinkercad module, students appeared to "acquire the knowledge and [some] skills needed to promote sustainable development" (UNESCO, 2016, p. 287) when they designed and created their 3D sustainable home with renewable resources and energy (see Figure 2). These design challenges could create a learning environment where students feel comfortable sharing their ideas and solutions to real-world problems. We describe these types of learning environments as "collaborative communities" where students "were motivated to explore many features by trying, asking questions, and sharing ideas with each other" (Instructor 3, Year 2). This is in line with Kafai and Burke's (2014) suggestion to shift from tools to communities (where the focus is on students sharing their ideas rather than the technology or digital tool). In further iterations of the STEAM camp (2022), students were able to explore different types of sustainable housing globally, such as examples noted in Ghana (butterfly roof to collect rainwater), Brazil (smart home with composting toilet), and Mexico (green roof to grow vegetation). In further iterations of the camp, we plan on providing students with more opportunities to talk about (i.e., engage in rich conversations that include in-depth knowledge and understanding) and explore these sustainable development goals in a real-world context.

Conclusion

This paper discussed how upper elementary school students develop a deeper understanding of mathematics, science, and coding through digital design. Our main findings were that students a) appeared to develop a deeper understanding of disciplinary and interdisciplinary concepts, b) seemed to engage more with the digital tools when they were remixing, improving, and reimaging the design, and c) applied the knowledge in the design projects to global competencies (simulations, animations, and gaming). According to OME (2020), "coding can support the development of a deeper understanding . . . [and] provide an opportunity for students to communicate their understanding of mathematical concepts" (p. 161). Our results showed how the students were able to learn mathematics, science, and technology concepts while applying their ideas to cross-curricular contexts (e.g., they learned engineering, science, and mathematics while designing their home in Tinkercad) and beyond the camps' curricular content (e.g., they learned about sustainable housing and that people's life choices and their own can have a positive impact on the environment). The STEAM camp approach demonstrates how to implement activities that allow students to explore, experiment, remix, and debug their code (Resnick & Rusk, 2020). It also demonstrates how the students can be given more freedom to code and design their own projects through the design challenges and open-ended activities, thus promoting a "low floor, high ceiling, and wide walls" (Gadanidis, 2015, p. 308) approach to learning. The findings of this research have implications for improvements in researching, designing, and implementing design projects as a pedagogical approach to teaching mathematics, science, and coding, as well as for optimizing learning in mathematics and other STEAM disciplines, in both in- and out-of-school contexts.

Acknowledgments

This work has been funded by the following agency: The Social Sciences and Humanities Research Council (SSHRC). The authors of this study acknowledge the contributions of the following undergraduate and graduate students in the curriculum development and facilitation of the activities: Celina Murray, Derek Tangredi, Frank Oliveira, Jade Roy, and Kelly Cluness.

References

Barichello, L. (2016). The movement towards a more experimental approach to problem solving in mathematics using coding. *International Journal of Mathematical Education in Science and Technology*, *47*(5), 791-797. https://doi.org/10.1080/0020739x.2015.1109147

Bertrand, M. (2019). *STEAM education in Ontario, Canada: A case study on the curriculum and instructional models of four K-8 STEAM programs* [Unpublished master's thesis]. Western University. https://ir.lib.uwo.ca/etd/6137/.

British Columbia Ministry of Education (BCME). (2016). *Applied design, skills, and technologies, K-12 curriculum* [Program of studies]. https://curriculum.gov.bc.ca/sites/curriculum.gov.bc.ca/files/curriculum/adst/en_adst_k-9_elab.pdf

Burke, Q., O'Byrne, W. I., & Kafai, Y. B. (2016). Computational participation: Understanding coding as an extension of literacy instruction. *Journal of Adolescent & Adult Literacy*, *59*(4), 371-375. https://doi.org/10.1002/jaal.496

Cobb, P., Confrey, J., DiSessa, A., Lehrer, R., & Schauble, L. (2003). Design experiments in educational research. *Educational Researcher*, *32*(1), 9-13. https://doi.org/10.3102/0013189X032001009

Colgan, L. (2020). Elementary math matters: Logo-coding to learn, not learning to code. *Ontario Mathematics Gazette, 59*(1), 35–37. https://www.proquest.com/docview/2448445666?pq-origsite=gscholar&fromopenview=true

Cotton, T. (2001) 'Mathematics teaching in the real world'. In P. Gates (Ed.), *Issues in mathematics teaching* (pp. 23-37). Routledge.

Council of Ministers of Education, Canada (CMEC). (2020). *Ensuring inclusive and equitable quality education: Sustainable development goal 4 in Canada* [Policy document]. https://www.cmec.ca/Publications/Lists/Publications/Attachments/407/Sustainable%20Development%20Goal%204%20in%20Canada%20EN.pdf

diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. MIT Press.

diSessa, A. A. (2018). Computational literacy and "The Big Picture" concerning computers in mathematics education. *Mathematical Thinking and Learning, 20*(1), 3-31.

Dohn, N. B. (2020). Students' interest in Scratch coding in lower secondary mathematics. *British Journal of Educational Technology*, *51*(1), 71–83. https://doi.org/10.1111/bjet.12759

Gadanidis, G. (2015). Young children, mathematics, and coding: A low floor, high ceiling, wide walls environment. In D. Polly (Ed.), *Cases on technology integration in mathematics education* (pp. 308-329). IGI Global. http://doi:10.4018/978-1-4666-6497-5.ch015

Gibbs, G. R. (2007). Thematic coding and categorizing. *Analyzing Qualitative Data*, *703*, 38-56.

Glaser, B. G., & Strauss, A. L. (2017). *Discovery of grounded theory: Strategies for qualitative research.* Routledge.

Goldenberg, E. P., & Carter, C. J. (2021). Programming as a language for young children to express and explore mathematics in school. *British Journal of Educational Technology*, *52*(3), 969–985. https://doi.org/10.1111/bjet.13080

Guest, G., MacQueen, K. M., & Namey, E. E. (2011). *Applied thematic analysis*. Sage.

Higginson, W. (2017). From children programming to kids coding: Reflections on the legacy of Seymour Papert and half a century of digital mathematics education. *Digital Experiences in Mathematics Education*, *3*(2), 71–76. https://doi.org/10.1007/s40751-017-0030-3

Hubert, T. L. (2014). Learners of mathematics: High school students' perspectives of culturally relevant mathematics pedagogy. *Journal of African American Studies*, *18*(3), 324-336. https://doi.org/10.1007/s12111-013-9273-2

Israel, M., & Lash, T. (2020). From classroom lessons to exploratory learning progressions: Mathematics + computational thinking. *Interactive Learning Environments*, *28*(3), 362–382. https://doi.org/10.1080/10494820.2019.1674879

Kafai, Y. B. (1996). Software by kids for kids. *Communications of the ACM*, *39*(4), 38-39. https://doi.org/10.1145/227210.227221

Kafai, Y. B. (2016). From computational thinking to computational participation in K-12 education. *Communications of the ACM*, *59*(8), 26-27. https://doi.org/10.1145/2955114

Kafai, Y. B., & Burke, Q. (2014). *Connected code: Why children need to learn programming*. Mit Press.

Ke, F. (2014). An implementation of design-based learning through creating educational computer games: A case study on mathematics learning during design and computing. *Computers and Education*, *73*, 26–39. https://doi.org/10.1016/j.compedu.2013.12.010

Miller, J. (2019). STEM education in the primary years to support mathematical thinking: Using coding to identify mathematical structures and patterns. *ZDM*, *51*(6), 915–927. https://doi.org/10.1007/s11858-019-01096-y

Nova Scotia Department of Education (NSDE). (2015). *Coding strategy* [Policy document]. https://www.ednet.ns.ca/psp/files-psp/codingstrategy.pdf

Ontario Ministry of Education (OME). (2007). *The Ontario curriculum grades 1-8: Science and technology* [Program of studies]. http://www.edu.gov.on.ca/eng/curriculum/elementary/scientec18currb.pdf

Ontario Ministry of Education (OME). (2020). *The Ontario curriculum grades 1-8: Mathematics* [Program of studies]. https://www.dcp.edu.gov.on.ca/en/curriculum/elementary-mathematics

Ontario Ministry of Education (OME). (2021). *Grade 9 math: A guide for parents* [Program of studies]. https://www.dcp.edu.gov.on.ca/en/multi-languages/english

Ontario Ministry of Education (OME). (2022). *Key changes - Science and technology, grades 1-8* [Program of studies]. https://www.dcp.edu.gov.on.ca/en/sci-tech-key-changes/download

Papert, S., & Harel, I. (1991). Situating constructionism. *Constructionism*, *36*(2), 1-11. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.691.4506&rep=rep1&type=pdf

Parker, F., Bartell, T. G., & Novak, J. D. (2017). Developing culturally responsive mathematics teachers: Secondary teachers' evolving conceptions of knowing students. *Journal of Mathematics Teacher Education*, *20*(4), 385-407. https://doi.org/10.1007/s10857-015-9328-5

Resnick, M. (1998). Technologies for lifelong kindergarten. *Educational Technology Research and Development*, *46*(4), 43-55. https://doi.org/10.1007/BF02299672

Resnick, M., & Rusk, N. (2020). Coding at a crossroads. *Communications of the ACM*, *63*(11), 120-127. https://doi.org/10.1145/3375546

Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-López, J. M. (2020). Computational thinking and mathematics using Scratch: An experiment with sixth-grade students. *Interactive Learning Environments*, *28*(3), 316-327. https://doi.org/10.1080/10494820.2019.1612448

Schell-Straub, S. (2013). Mathematics education meets development education: The competency' mathematical modelling' combined with global skills and competencies in a secondary school project in Germany. *International Journal of Development Education and Global Learning*, *5*(1), 7-31. https://doi.org/10.18546/IJDEGL.05.1.02

Strauss, A., & Corbin, J. M. (1997). *Grounded theory in practice*. Sage.

Sung, W., Ahn, J., & Black, J. B. (2017). Introducing computational thinking to young learners: Practicing computational perspectives through embodiment in mathematics education. *Technology, Knowledge and Learning*, *22*(3), 443-463. https://doi.org/10.1007/s10758-017-9328-x

United Nations Educational, Scientific and Cultural Organization (UNESCO). (2016). *Education for people and planet: Creating sustainable futures for all* [Global education monitoring report]. UNESCO publishing. https://uis.unesco.org/sites/default/files/documents/education-for-people-and-planet-creating-sustainable-futures-for-all-gemr-2016-en.pdf

Corresponding Author Contact Information:

Author name: Marja Gabrielle Bertrand

Department: Curriculum Studies

University, Country: Western University, Canada

Email: mgbertra@uwo.ca

Please Cite: Bertrand, M. G., Namukasa, I. K., & Li, L. (2023). STEAM Camp: Teaching Middle School Students Mathematics, Science and Coding through Digital Designs. *Journal of Research in Science, Mathematics and Technology Education*, 6(SI), 47-67. DOI: https://doi.org/10.31756/jrsmte.213SI

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Ethics Statement: This study was reviewed and approved by the research ethics permitting board at Western University.

Author Contributions: All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Received: March 01, 2023 ▪ Accepted: May 23, 2023