Can. J. Sci. Math. Techn. Educ.
https://doi.org/10.1007/s42330-024-00330-4

Developing Computational Thinking Using LYNX for Loom Beading Designs in Grade 5

Ruth Beatty[1] · Colinda Clyne[2] · Leslie-Anne Muma[3] · Jennifer Parkinson[3] · Bonnie Sears[4]

Accepted: 15 July 2024

Abstract In this study, a research team made up of Métis artists and knowledge keepers, Anishinaabe and non-Indigenous educators, and a non-Indigenous university mathematics education researcher co-designed and delivered an Indigenous cultural mathematical inquiry in a Grade 5 classroom. We explored the connections between loom bead designs and computational thinking of students using the LYNX coding platform to create and refine bracelet designs. Results indicate this investigation supported the development of (1) incremental and iterative approaches to coding, (2) testing and debugging, (3) reusing and remixing code, and (4) abstracting and modularizing. More importantly, students gained an appreciation of Métis culture and the importance of experiencing multiple cultural perspectives when learning mathematics.

Résumé Dans cette étude, une équipe de recherche composée d'artistes et de gardiens du savoir métis, d'éducateurs anichinabés et allochtones, ainsi que d'un chercheur universitaire non autochtone spécialisé dans l'enseignement des mathématiques a conçu et réalisé conjointement une enquête mathématique culturelle autochtone dans une classe de cinquième année. En utilisant la plateforme de codage LYNX, nous avons exploré les liens qui peuvent exister entre les motifs de broderie perlée et la pensée computationnelle des élèves dans la création et l'amélioration de modèles de bracelets. Les résultats indiquent que cette enquête a appuyé l'élaboration 1) d'approches incrémentielles et itératives en matière de codage, 2) de tests et de processus de débogage, 3) d'approches pour réutiliser

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s42330-024-00330-4.

✉ Ruth Beatty
rbeatty@lakeheadu.ca

[1] Faculty of Education, Lakehead University, Orillia, ON, Canada

[2] Anishinaabe Kwe, Upper Grand District School Board, Kitgan Zibi Anishinabeg, Guelph, ON, Canada

[3] Red River Métis Artisan and Knowledge Keeper, Guelph, ON, Canada

[4] Upper Grand District School Board, Guelph, ON, Canada

Published online: 03 September 2024

et remanier les codes, et 4) de processus d'abstraction et de modularisation. Surtout, les élèves ont appris à apprécier la culture métisse et l'importance d'éprouver de multiples perspectives culturelles lors de l'apprentissage des mathématiques.

Keywords Culturally responsive mathematics · Computational thinking · LYNX

Introduction

As part of an initiative to teach math through First Nations and Métis cultural practices, the first author has worked in partnership with Anishinaabe, Métis, and Cree community partners across Ontario to explore the connections between loom beading and computational thinking by integrating computer programming with Indigenous beadwork. This is part of a larger multi-year, multi-site study where research teams made up of Indigenous artists and knowledge keepers and non-Indigenous educators have collaborated to prioritize Indigenous knowledges when teaching and learning mathematics.

The impetus for this work is a recognition that Canada's school curricula have historically prioritized European educational philosophies, world views, and content. Indigenous Peoples have been on these lands for millennia, "since time immemorial", and so the goal of this work is to build relationships between community partners and non-Indigenous educators in order to incorporate Indigenous ways of knowing and pedagogical approaches into the math classroom. National and provincial mandates require the integration of Indigenous perspectives, world views, and contributions across K-12 teaching (e.g. Ontario Ministry of Education, 2020; Truth and Reconciliation Commission of Canada (TRC), 2015) to support identity building for Indigenous students, and an increased knowledge and appreciation of Indigenous perspectives and values from non-Indigenous students and educators. Fundamental shifts in education call for educators to engage in consultation, collaboration, consensus building, and participatory research in collaboration with Indigenous peoples (Battiste et al. 2002).

This work investigates the mathematics inherent in different Indigenous cultural practices. The goal is to make complex math concepts more relatable for Indigenous and non-Indigenous students by contextualizing them in hands-on investigations. Research teams have explored the mathematics inherent in art forms such as different types of beading, birch bark basket making, and moccasin making. The project described in this paper is a collaboration among two Red River Métis Knowledge Keepers and artists, the Coordinating Principal of Indigenous Education at a provincial school board who is Anishinaabe, two non-Indigenous educators, and a university researcher to explore the connections between loom beading and mathematics. All of the work for this project, from goal setting to dissemination, was conducted in partnership.

Although the mathematics of looming is documented in previous publications (Beatty, 2018; Beatty and Ruddy 2019; Beatty & Blair, 2015), for this project, we were interested in incorporating coding as part of the design process and to explore connections among design, beading, coding, and computational thinking. This was in part inspired by one of the initial conversations the first author had at the beginning of this work on decolonizing mathematics instruction when speaking with Elder Stephen Kejick of Iskatewizaageegan First Nation in 2012. Kejick spoke about his vision of Indigenous Education as seeing Indigenous students walking with "a feather in one hand and a computer in the other".

The project was also a response to changes in the provincial math curriculum. Formal study of computational skills in primary schools became part of the Ontario Math curriculum in 2020, but much of coding tends to be taught independently as a stand-alone activity during province-wide initiatives such as the "hour of code" (https://code.org/). Teaching coding is not only about teaching children how to code but also about teaching problem solving using logic and creativity, ideally in an environment that fosters risk-taking, experimentation, and provides a meaningful context.

In this paper, we use the term Indigenous to name the Inuit, Métis, and First Nations peoples of what is now Canada, although we recognize that the word Indigenous is problematic and does not account for the unique nature of each First Nation, Métis, or Inuit community. Anishinaabeg refers to a group of culturally and linguistically related First Nations concentrated around the Great Lakes. Métis are one of the distinct Indigenous Peoples of Canada whose family lines can be traced to the historic Métis Nation homelands in west-central North America. Métis are distinct from their First Nations or European ancestors and have a unique culture, tradition, language, and world view. The Métis National Council adopted a definition of Métis at the General Assembly in 2002 as "a person who self-identifies as Métis, is distinct from other Aboriginal [Indigenous] peoples, is of historic Métis Nation Ancestry and who is accepted by the Métis Nation" (Dubois & Saunders, 2017, p. 887).

Methods

As outlined in a previous publication (Beatty & Clyne, 2020), the authors identified relational protocols from Indigenous knowledge systems to build long-term relationships with community that are grounded in respect, relevance, reciprocity, responsibility, and relationships (Tessaro et al., 2021; Archibald et al., 2019; Kirkness & Barnhardt, 1991). Through the processes of co-planning, co-teaching, and co-reflecting, this work has been about lifting up Indigenous knowledge and shifting research practices towards an emphasis on ethical relationality (Donald, 2012). In this work, we seek "to understand how our different histories and experiences position us in relation to each other" (Donald, 2012, p. 535). We are exploring new ways of teaching math that are led by the cultural knowledges of community artists. It is important to note that the researcher and participating teachers, all of whom are non-Indigenous educators, had spent years building relationships with the Métis artists because embarking on this type of cultural mathematical investigation can only be successful when led by community within a trusting, respectful, and reciprocal partnership. These relationships were facilitated by the school board level Coordinating Principal of Indigenous Education who was instrumental in bringing this project to life. Using traditional Indigenous practices like looming to teach coding (an integral component of STEM education) is potentially problematic given the current push in STEM subjects (Science, Technology, Engineering, and Mathematics) towards curriculum initiatives that promote consumerism and capitalism (Donald, 2019; Wiseman, et al., 2020); however, we were interested to investigate teaching coding and computational thinking within a context that centres Indigenous ways of knowing, being, and doing.

Our research question was "What computational thinking and coding skills might students have the opportunity to learn through Indigenous design and artistry?" All lessons were video recorded, and field notes were completed each day by the first author. These notes included an overview of the sequence of activities and a summary of the mathematical thinking evidenced through students' verbal answers, written work, or coded procedures. After each lesson, the artists, teachers, and researcher debriefed about the day's findings and discussed lines of inquiry for the following day. Throughout the project, interviews with the students and Métis artists were video recorded to capture their thinking, impressions of the work, and the cultural connections they perceived.

We present this 2-week looming and coding investigation as an example of how coding can be meaningfully taught through Indigenous culture and cultural practices. The project was carried out in a Grade 5 classroom in a provincially funded school in Southern Ontario and led by the Métis community partners, Leslie and Jennifer, both of whom are artists and expert beaders. They began by teaching some of the history of the Métis people. They showed the students different examples of Métis beadwork and discussed Métis and Anishinaabe motifs, which are typically floral

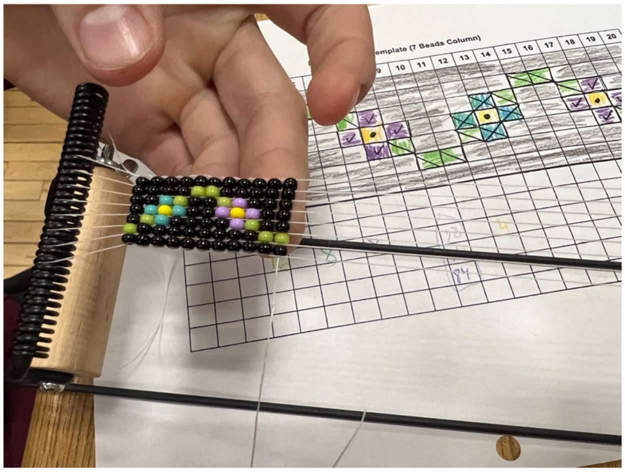


Fig. 1 An example of loom beading using Leslie's design. Beadwork showing a floral design on a loom. Eight horizontal threads run from one coiled end of the loom to the other, and columns of beads are strung in between the horizontal threads

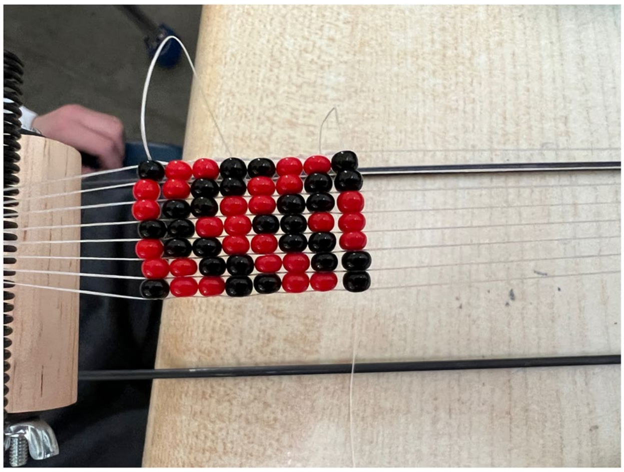


Fig. 2 An example of looming a chevron design. Beadwork showing a chevron design of alternating black and red chevrons on a loom. Eight horizontal threads run from one coiled end of the loom to the other, and columns of beads are strung in between the horizontal threads

designs inspired by nature or more geometric and abstract. Integrated within these teachings were Métis instructional traditions, which included themes such as respecting the materials (especially the beads), recognizing individual student's strengths, and pooling strengths to ensure everyone is successful.

Looming and Coding

Leslie and Jennifer introduced looming to the students. Looming is a type of beading that is done on a loom and involves stringing beads onto vertical weft threads and weaving them through horizontal warp threads (Figs. 1 and 2).

Leslie and Jennifer explained that although Métis are known for their beadwork, looming is not strictly a Métis practice, but was taught to them by their First Nations relations and neighbours. Leslie showed the students how she designed her beadwork using graph paper, and then introduced a template made up of 7 rows and 20 columns. The columns of the template represent the columns of seven beads that are strung onto weft threads and woven into the warp threads. They introduced the students to

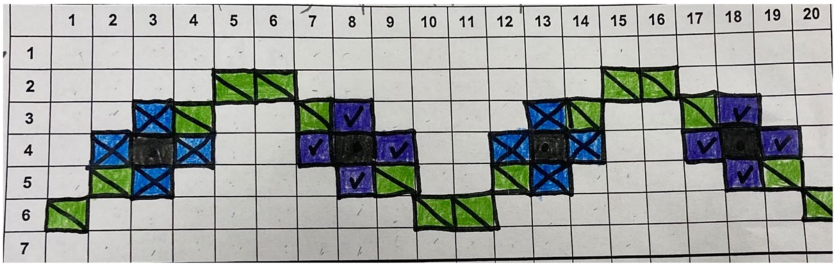


Fig. 3 Design template with Leslie's Métis flower looming design. A floral pattern coloured on a 20 columns × 7 row template

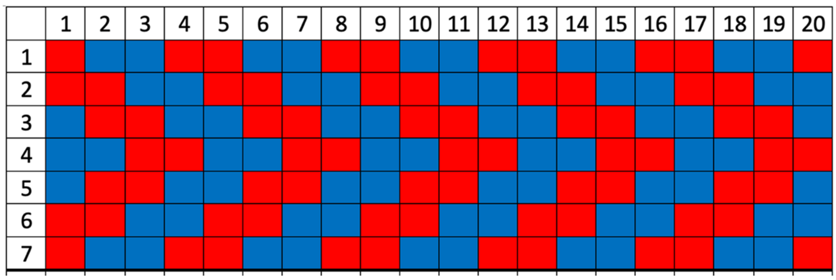


Fig. 4 Design template with chevron design. A chevron pattern coloured on a 20 columns × 7 row template

two different designs, one a flower design that Leslie designed based on traditional Métis floral motifs interpreted within the grid of a looming template (Fig. 3), and the other was a geometric chevron design (Fig. 4), which is seen both in Métis finger weaving and also in Anishinaabe loom beading.

Leslie and Jennifer taught students to loom by following the columns of the template and stringing each column of beads to weave into their bracelets.

Coding the Chevron Design

At this point, we wondered if we could transform the design process from colouring on a static paper template to creating a dynamic representation of looming patterns using computer coding, and how this might support the development of students' computational thinking. Although there are many definitions for computational thinking, in our work, we consider it as the problem-solving cognitive process that allows students to formulate a problem in such a way that it can be solved by writing code (Papert, 1980; Wing, 2008; Cuny et al., 2010). To foster creative problem solving, we chose to use the LYNX coding program, which is a text-based program based on the LOGO turtle-geometry programming language created by Seymour Papert that offers students the opportunity to create their own coding procedures based on 44 "turtle commands". LYNX provided an additional way to investigate the patterns within looming designs and, in the process of creating the code for these designs, the related patterns within the pattern coding procedures. The setup of LYNX was perfect for this because it incorporates both the text of the code and also shows the visual representation of the code (i.e. the beadwork pattern) at the same time. This was an unfamiliar program for everyone involved in the project. Many of the students had used block-based coding programs such as Scratch, but this was their first time learning how to write and execute their own coding procedures. It was also a new program for the teachers and for Jennifer and Leslie. This project represents true inquiry within a community of learners.

We began by teaching the students some of the commands the turtle understands. As we introduced each new command, we added it to a table that students could use as an anchor chart during our investigations.

Command	Shortcut	What does the turtle do?
forward [number]	fd [number]	Moves forward the number of pixels you input Forward 100 means the turtle will move forward 100 pixels
back [number]	bk [number]	Moves backward the number of pixels you input Backward 100 means the turtle will move backward 100 pixels
right turn [angle]	rt [angle]	Turns the number of degrees you input Right turn 90° means the turtle will turn right 90°
left turn [angle]	lt [angle]	Turns the number of degrees you input Left turn 90° means the turtle will turn left 90°
pen up	pu	Will not create a line when moving around the screen
pen down	pd	Will create a line when moving around the screen
setpensize [number]		Adjusts the thickness of the line according to the number of pixels indicated. Setpensize 15 creates a line 15 pixels wide
setcolour [colour name] or setcolour [colour number]	setc	Change the colour of the line drawn. The turtle understands 16 colour names, and also knows 139 colour numbers associated with 139 different colours
clear graphics	cg	Erases any images on the screen and returns to the centre of the screen

We also taught the students the syntax for creating a procedure, including beginning each procedure with "to [name of the procedure]" and ending with the word "end". Once we had done a bit of exploration with the program, we posed this problem to the students, "how might we use LYNX to create looming designs?" To begin, we chose to focus on programming the turtle to create the simpler chevron design rather than the more complex floral pattern.

The resource teacher, Bonnie, led a whole class discussion about how they might use the LYNX commands to create a virtual bead. Using their knowledge of drawing lines of different thicknesses and colours, the students and the teacher co-constructed a procedure for creating a redbead by setting the pen size to 15 pixels, changing the colour of the line to red, and having the turtle move forward 15 pixels to create a virtual square bead of 15 × 15 pixels (Fig. 5).

The students then used this information to create a blue bead by copying the *redbead* procedure and changing two things, the name of the procedure from *redbead* to *bluebead* and the colour of the 15-pixel by 15-pixel line. Many students discovered they could simply copy and paste the *redbead* procedure and alter it to create the *bluebead* procedure. In Video 1, a student, E, explains his process of copying and pasting his *redbead* code to create a *bluebead* code.

Initially, students tried to manually stack individual beads in crooked columns (Fig. 6).

Fig. 5 The *redbead* procedure. A LYNX procedure that reads from line 1 to line 6: to redbead, setpensize 15, setcolour red, pendown, forward 15, end

```
1 to redbead
2 setpensize 15
3 setcolour 'red
4 pendown
5 forward 15
6 end
```

Fig. 6 Manually positioning the turtle to create each bead in a column. Small squares of red and blue, representing red and blue beads, awkwardly placed in vertical columns of seven squares each. There are gaps between each small square and the columns are crooked

One student, D, shared her discovery that if she ran the *redbead* and *bluebead* procedures in the correct order one after the other, the turtle would stack the beads. This mimicked the way beads are stacked in columns during looming. D demonstrated how she created the first two columns of the chevron design by having the turtle stack the beads for each column and then physically moving the turtle each time she wanted to create the next column (Video 2).

The students adopted this method of creating the columns of the pattern by stacking the virtual beads, but soon became frustrated with having to keep typing out the name of the bead procedure for each individual bead to create the stack of beads for each column. One student, J, figured out that he could create a *column1* procedure so that the turtle would stack the beads in the same order as in the first column on the template. Instead of having to type "redbead, redbead, bluebead, bluebead, bluebead, redbead, redbead" to create the column, J simply had to type "column1" (Fig. 7). He then created a *column2* procedure by listing all of the beads in order from bottom to top in column 2. The classroom teacher asked J to share his idea with the rest of the class. This led to a discussion about creating a "program inside a program" in order to be more efficient with coding.

To make their coding even more efficient, we introduced students to the "repeat" command, so nine lines of code for column 1 could be shortened to five (Fig. 8).

The next challenge was trying to perfectly align each new column to the right of the previous column (again, mimicking the beading process). In Video 3, we see the difficulty. The students had to manually move the turtle down the screen to create the next column.

The solution to this problem was trickier and required the students to think about the movement of the turtle from the turtle's perspective. As a whole class, we talked through the problem of how to get the turtle from the top of one column to the bottom of the next column, ready to stack the specific sequence of beads which represented the way students loomed by always stringing each column of beads

Fig. 7 J's *column1* procedure. A LYNX procedure that reads from line 16 to line 24: to column 1, redbead, redbead, bluebead, bluebead, bluebead, redbead, redbead, end

```
16 to column1
17 redbead
18 redbead
19 bluebead
20 bluebead
21 bluebead
22 redbead
23 redbead
24 end
```

Fig. 8 Modifying the *column1* procedure using the repeat command. A LYNX procedure that reads from line 16 to line 20: to column 1, repeat 2 redbead, repeat 3 bluebead, repeat 2 redbead, end

```
16 to column1
17 repeat 2 [redbead]
18 repeat 3 [bluebead]
19 repeat 2 [redbead]
20 end
```

by reading the template bottom to top. Students then worked on their own or with a partner to generate a number of ideas (all of which were based on the same turtle movements but executed in different orders).

As students brainstormed how to get the turtle to move down the column and over to the right, they discovered that as the turtle moved down the column of beads, the entire column turned red! This was unexpected and required students to revisit their original bead procedures. One student, M, discovered that her original bead procedures did not include a "penup" command after the beads were created, so when she was exploring how to move the turtle down the column of beads, it continued drawing a red line. Her solution required modifying the original redbead and bluebead procedures to include the "penup" or "pu" command prior to ending the procedure. To move the turtle to the next columns, M used the "back" command to move the turtle back 105 pixels, or "turtle steps", which she calculated by knowing each virtual bead was 15 pixels high, and there were 7 beads in the column. She then had the turtle turn right 90 degrees, move forward 15 pixels (the width of a bead) and then left turn 90 degrees to be ready to create the next column.

Another solution was to begin by having the turtle turn right 90 degrees, move forward 15 pixels, turn right another 90 degrees, move forward 105 pixels, and then turn right (or left) 180 degrees. A third was to have the turtle make a right (or left) turn of 180 degrees, move forward 105 pixels, turn left 90 degrees, move forward 15 pixels, and then make another left turn of 90 degrees. This latter solution required the most "in the turtle's shell" thinking (a term coined by the students) because if the turtle faces the bottom of the screen, left turns will result in facing to the right of the screen.

Once students had modified their bead procedures, the *move* procedure was inserted into each of the *columns* procedures, again providing an opportunity for students to experience modifying existing code (Fig. 9).

Fig. 9 Modifying the original column procedures to include the *move* procedure. The column 2 procedures modified to include the *move* sub-procedure before end

```

to column2
bluebead
repeat 2 [redbead]
bluebead
repeat 2 [redbead]
bluebead
move
end

to move
bk 105
rt 90
fd 15
lt 90
end
```

Initially, many students created many column procedures (some students created up to 20 different column procedures because there were 20 columns on the template). This led to an interesting discussion about how many column procedures were actually needed to create the chevron design and if students could see any repetitions in the procedures they created. They immediately saw that the procedure for creating column 5 was the same as for creating column 1, the procedure for creating column 6 was the same as for creating column 2, and so on. This led to a discussion about how we could more efficiently create repeating designs by creating a *core* procedure, which essentially told the turtle how to build the unit of repeat, or pattern core, using the first four column procedures. The new procedure meant that, instead of running the column procedures over and over again, students could use the column procedures as sub-procedures in a new super-procedure called "core". Not only was this a more efficient way to model a full bracelet, it also supported students to identify which columns made up the unit of repeat.

In Video 4, we see an example of a student who played with the idea of reflecting her core. Originally, her *reflectcore* procedure was the same as her *core* procedure, and she was unsure how to get the turtle to reflect the core. She turned the template upside down to see what the reflection would look like and then realized that she would have to begin with column 4 rather than column 1, and so amended her procedure by entering columns 4 through 1. Other students began playing with the cores and reflections to create bracelet designs with one or multiple vertical lines of reflection. They also began playing with different colours of beads (Video 4 chevron reflections).

Designing Looming Patterns Using *LYNX*

During our final investigation, we invited students to create a 4-column pattern core that did not have a vertical line of symmetry by colouring 4 columns on the template. Once only the four columns were coloured, we asked them to code the core and then play with different lines of reflection, using the chevron core as an example (Fig. 10).

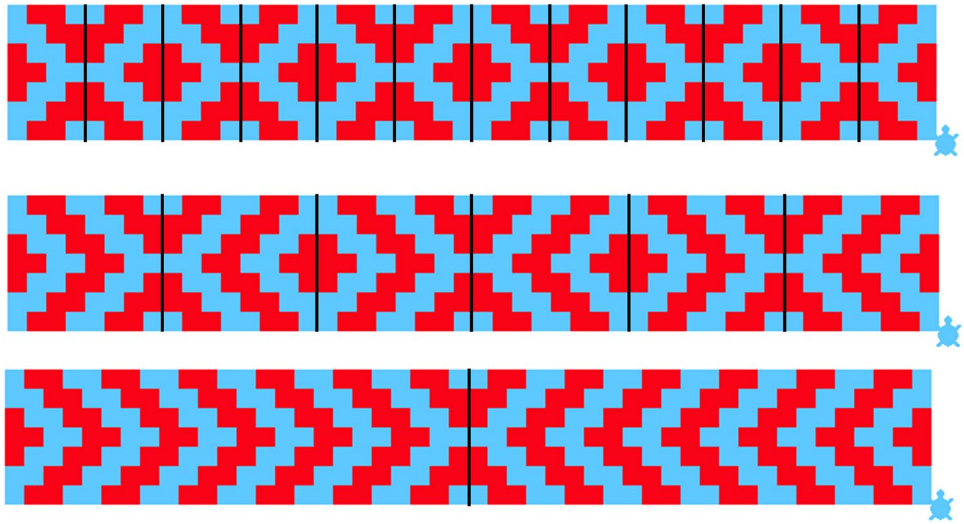


Fig. 10 The chevron pattern core with different lines of reflection [lines of reflection have been added for clarity]. The top pattern has alternating pattern cores and reflected cores (eleven lines of reflection), the middle pattern has alternating double cores and double reflected cores (five lines of reflection), and the bottom pattern has six cores and then six reflected cores (one line of reflection)

The students were delighted by the different complex patterns they created based on a simple pattern core and varying lines of symmetry. Students experimented with different designs, as seen in these examples (Fig. 11).

Once they had played with at least three different design ideas, the students chose one to bead. In order to determine how long to make their bracelets, Leslie taught the students about the proportions they were working with. For these projects, the students worked with size 8 beads, which meant that 4 beads were equal to about 1 cm. Leslie connected this to the template by showing the students how four columns on the template represented about 1 cm of beadwork. She asked students to measure their wrists and then subtract 4 cm for the 2 cm hide ends on each side of the bracelet. Students then used the remaining number of cm to calculate how many cm of beadwork they would need, how many columns they would need, and how many repetitions of the core (reflected or not reflected) they could repeat.

Results

Computational Thinking

In this work, we identified four computational thinking practices that naturally arose from the investigations (Brennan and Resnick, 2012): being incremental and iterative, testing and debugging, reusing and remixing, and abstracting and modularizing.

Incremental and Iterative

This practice refers to the fact that developing coding procedures in small steps—first by imagining what they want the turtle to accomplish, developing a procedure, trying it out, refining it, and developing it further. Students used incremental and iterative approaches to develop the different components

Fig. 11 Experimenting with different lines of reflection—coding and beading. Some examples of student designs using various combinations of lines of reflection. This figure shows the LYNX designs and the corresponding beadwork

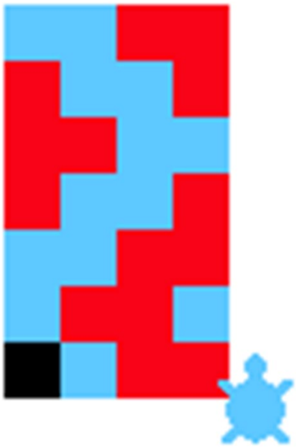
of the bracelet design beginning with procedures for beads and then using those experiences to create procedures for columns, cores, and reflected cores. Through the iterative nature of the text-based LYNX program, students investigated the structure of the patterns of looming designs, particularly finding the unit of repeat of the pattern by discovering they have to only code four columns of beads and then use those to create a core procedure and units of repeat within the pattern. For example, some students noticed the horizontal lines of reflection within each of the columns of the chevron designs, or that the first and third columns and second and fourth column were the same but with the colours switched (2 red, 3 blue, 2 red vs 2 blue, 3 red, 2 blue). Creating beads, columns, cores, and bracelet procedures allowed students to understand the structure of looming designs more deeply and identify patterns both within the design and within the structure of each procedure, and how the two were connected. They used these patterns to continuously refine their code to make it more efficient, so that they could use these more efficient and smaller procedures to build more and more complex super-procedures.

During an interview about the project, Leslie cited the incremental and iterative nature of the process of learning more and more complex coding as fundamental to student success. “Going step by step helped a lot. Students were breaking it down to smaller pieces and building those into bigger procedures so that they weren’t being overwhelmed.”

Testing and Debugging

This practice refers to the kinds of testing and debugging practices students develop as they create their codes and include reading through procedures, experimenting with different procedures, looking for typographical errors, and imagining each step of the procedure being carried out by the turtle. One of the best things about using LYNX to support the development of problem solving and critical thinking was the fact that students received immediate feedback from the software if their procedures did not work as they anticipated. Very often, this was due to a syntax error—a misplacement of spaces or capital letters or misspelling of procedure names or commands, but sometimes, this was due to imperfections in the code. For example, when writing a column procedure, we could see that as some students created their first few columns, the columns did not look like they should (Fig. 12). It took a whole class discussion to debug the code and recognize that although all components of the bead procedure were there, the order was incorrect. The students had written the “setcolour” command *after* the forward command, which meant that when the turtle went forward to create the square bead it was using the colour of the previous bead. Instead of red, red, blue, blue, blue, red, red for the first column, the turtle instead made black (the default colour of the turtle), red, red, blue, blue, blue, red. This emphasized the importance of

Fig. 12 Core created when the forward command came *after* the setcolour command in each bead procedure. A LYNX pattern core of four columns. The core is off by one bead in each column and so does not look like a chevron



"in the shell thinking" as students had to figure out the sequence of steps the turtle took to create a bead, and how creating a column relied on the correct sequence of the commands when making each bead.

Such misfires in the production of their designs evolved from being a source of frustration to a source of investigation and problem solving either independently or as a group. As students started to become more curious about the possibilities of LYNX, they discovered new ways of enhancing their coding, for example, creating buttons to make experimenting with lines of reflection easier. And further investigations led to further refinements in design, for example, what happens when we turn the turtle 180 degrees and then execute the core procedure? How can we play with horizontal lines of reflection to create a variety of designs based on the same 4-column core (Fig. 13)?

Another form of debugging happened when students made connections between the text-based code and the virtual design and checked whether it aligned with their template and/or their looming. If the two representations were different, the students would go back and figure out how to alter the code so that the virtual pattern was the same as the hand-coloured design.

Fig. 13 Playing with horizontal lines of reflection. These three examples are similar to the examples in Fig. 10 but include a horizontal line of reflection creating designs of 14 rows instead of 7 rows

Reusing and Remixing

This refers to the practice of building on previous work or building on each other's procedures and solutions to problems. We saw a lot of reusing and remixing of codes, particularly as students created their own unique bracelet designs. Students modified their initial bead procedures and column procedures in a number of ways, (1) by altering the colour of the beads, (2) by altering the order in which beads were stacked in each columns, and (3) for those students who wanted to be able to show the entire length of their bracelet on the LYNX screen, which is approximately 800 pixels long and 440 pixels wide, reduced the size of their bead to a 10 pixel by 10 pixel square and so had to adjust their move procedure. However, the structure of the *bead* and *column* procedures remained basically the same. The *core* and *reflectcore* procedures were copied as they were first written.

Students also built on other people's work as students shared their solutions for particular problems. For example, students learned from other students how to copy and paste procedures, how to modify existing procedures, and how to create "move" procedures to align columns within the core. The students also shared their discoveries as they experimented beyond the initial commands they were taught, for example, creating beads of different colours or changing the background colour of the screen.

Abstracting and Modularizing

This is defined by Brennan and Resnick as "building something large by putting together collections of smaller parts" (Brennan & Resnick, 2012, p. 9). As students solved the overarching question of "how do we use LYNX to design loom beading patterns?" we noted that students created algorithms to solve, at first, specific problems but that these algorithms were then generalized beyond particular instances. For example, learning how to create beads led to students being able to create beads of any colour. Learning to create columns and cores specifically for the chevron pattern led to creating codes for any design students could imagine. And as a result of learning to create codes for pattern cores and reflected cores, students learned to create designs through playing within reflections that they may not have initially thought of.

Discussion

The goal of this project was to work with Métis community partners, teachers, and students to realize the potential of integrating coding (specifically a text-based flexible coding platform) into our work connecting Indigenous cultural practices and mathematics instruction. At the end of the project, we identified four specific areas of student learning:

1. Problem solving through iteratively creating, reusing, remixing, debugging, and generalizing code;
2. Understanding the logic of writing procedures and of creating a series of nested procedures;
3. Increased mastery of writing code based on extended investigations within a meaningful context;
4. The creation of multiple connections among representations of beadwork design including concrete (beads), visual (templates/LYNX patterns), and abstract (code).

Students engaged in coding investigations that were meaningful and purposeful. They were not coding for the sake of coding; they were coding to create and test out more and more complex designs that they could then turn into loomwork bracelets. They also made connections between the symmetrical design options they played with and the importance of symmetry in Métis and Anishinaabe beadwork. This aligns with a constructionist approach to learning that highlights both the importance of solving problems using coding within a meaningful context (Papert, 1980) and the importance of students engaging in the development of artefacts (Kafai & Resnick, 1996). Truly, this was a journey of discovering the beauty of math.

Students' responses to the project were overwhelmingly positive. As Leslie remarked, "the students really enjoyed it, and once they got it going, they just took off!" The students spent hours writing hundreds of lines of code to experiment with their bracelet designs. Because LYNX is a cloud-based platform, we discovered that many students chose to continue to create codes for beading designs outside of class time either after school or on the weekend and would report back about the number of hours they spent coding or the number of lines of code they created.

One of the most important aspects of this work is creating reciprocal relationships with community artists and supporting their recognition of their own mathematical thinking (Beatty & Clyne, 2020). When discussing the project with Leslie, she talked about how much she enjoyed sharing her design expertise and using her designs as the impetus for learning. She said that both co-leading the project and learning together with the students were important parts of building relationships. "Being one-on-one and learning with the kids was great for me because I could associate myself with what they were learning. And even talking to them, you get a conversation going, so you're both learning and both thinking".

Although Leslie had never done any coding before, she said that she could see herself using LYNX as part of her design process in the future. "It's fun to see how doing all that coding, making your core and just hitting core, core, core and then having the bracelet represented on the screen. It's a lot of work to enter it all, but it's really nice to see it when it's done. I think I will use it [LYNX] more because a lot of times I design and then make the bracelet and then I'll see that I don't like the beads in a certain spot, whereas this way I'd see the whole pattern without having to bead it and then take it apart to start over."

She then shared the impact the project has had on her.

> It's a great learning experience. I had a lot of great moments with the kids as I learned with them. I kind of got stronger in my own understanding and I made relationships. And…I can't say enough about it because it has opened me up a lot. I'm not as shy as I was. It's been an overwhelmingly positive experience for me. I think anyone would enjoy it, whether they're a math teacher, whether they're someone like me who enjoys a craft and I know now that there's math that goes into everything! I came out with a better understanding of myself, and I know that I brought knowledge and cheer to the students I worked with.

Indigenous Practices and STEM

Although we saw computational thinking meaningfully developed in this project, we recognize that an area of tension in this work is the appropriateness of using coding to design beading patterns, which may have an implication of, for example, mass producing designs and beadwork. Leslie spoke to the students about how personal beadwork is to her, and that her design was hers alone, but that she had given permission for the students to use it as an initial point of investigation. She emphasized that each student needed to create their own designs, using colours and images that were meaningful to them rather than looking for inspiration from, for example, other artists' designs on the internet.

> Each persons' design is meaningful to them, it's important to them. It's really inappropriate to use someone else's design without their permission because their design is a part of them, just like your design is a part of you. I've given you permission to use my design so I could share with you some of the things that are important to me and so you could learn more about Métis people.

An approach to teaching coding that is rooted in world views that prioritize relationships and personal expression in design offers a counter approach to learning coding as a step towards commodification and consumerism. In our work, we aim to move "beyond concepts of STEM that privilege neoliberal imperatives of growth, innovation, and economics" (Wiseman et al. 2020).

We hope that this example of our work inspires others to explore connections between Indigenous art, design, and technology and elementary mathematics instruction, specifically, computational thinking and coding. However, we would like to reiterate that for non-Indigenous educators, projects like this *must be done in partnership with Indigenous artists*, *community partners, and knowledge keepers*. To do otherwise would defeat the purpose of this work, which is ultimately about building strong reciprocal and respectful relationships in order to prioritize Indigenous voices in mathematics education.

Funding This work was funded by a CanCode grant through Code to Learn.

Declarations

Competing Interests The authors declare no competing interests.

References

Archibald, J., Nicol, C., & Yovanovich, J. (2019). Transformative education for aboriginal mathematics learning: Indigenous storywork as methodology. In J. Archibald, J. Lee-Morgan & J. De Santos (Eds.), Decolonizing research Indigenous storyworks as methodology (pp. 72-88). London, UK: Zed Books.

Battiste, M., Bell, L., & Findlay, L.M. (2002). Decolonizing education in Canadian universities: An interdisciplinary, international, Indigenous research project. *Canadian Journal of Native Education 26,* (2): 82-95.

Beatty, R., & Clyne, C. (2020). Relationships and Reciprocity Towards Decolonizing Mathematics Education. *Journal of Higher Education Theory and Practice*, *20*(7). https://doi.org/10.33423/jhetp.v20i7.3156

Beatty, R. & Ruddy, C. (2019). Mathematics exploration through Algonquin beadwork. In M. Sack (Ed.) *My Best Ideas*, Rubicon.

Beatty, R., & Blair, D. (2015). Indigenous pedagogy for early mathematics: Algonquin looming in a grade 2 math classroom. *The International Journal of Holistic Early Learning and Development*, *1*, 3-24.

Beatty, R. (2018). Connecting Algonquin loomwork and Western mathematics in a Grade 6 math class. In T. Bartell (Ed.), *Toward Equity and Social Justice in Mathematics Education*, Springer. https://doi.org/10.1007/978-3-319-92907-1_7

Brennan, K. &Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Paper presented at the Annual Meeting of the American Educational Research Association, Vancouver, April 13–17, 2012.

Cuny, J., Snyder, L. & Wing, J.M. (2010). Demystifying computational thinking for noncomputer scientists. Unpublished manuscript in progress, referenced in http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf and in Wing, J. (2011), Research notebook: Computational thinking—What and why. The Link Magazine (6), 20-23.

Donald, D. (2012). Indigenous metissage: A decolonizing research sensibility. *International Journal of Qualitative Studies in Education, 25(* 5), 533-555. https://doi.org/10.1080/09518398.2011.554449

Donald, D. (2019). Homo economicus and forgetful curriculum. *Indigenous education: New directions in theory and practice*, 103–125.

Dubois, J., & Saunders, K. (2017). Rebuilding Indigenous nations through constitutional development: a case study of the Métis in Canada. *Nations and Nationalism, 23*(4), 878-901.

Kafai, Y. B., & Resnick, M. (1996). *Constructionism in practice: Designing, thinking, and learning in a digital world.* Hillsdale, NJ: Erlbaum.

Kirkness, V. J., & Barnhardt, R. (1991). First Nations and higher education: The four Rs-respect, relevance, reciprocity, responsibility. *Journal of American Indian Education, 30(3),* 1-15.

Papert, Seymore. (1980). *Mindstorms: Children, Computers, and Powerful Ideas*. New York, NY: Basic Books.

Tessaro, D., Restoule, J.-P., Gaviria, P., Flessa, J., Lindeman, C., & Scully-Stewart, C. (2021). The five R's for Indigenizing online learning: A case study of the First Nations school's principals course. *The Canadian Journal of Native Education, 40* (1), 125-143. https://doi.org/10.14288/cjne.v40i1.196606

The Ontario Curriculum, Grades 1–8: Mathematics, (2020). Queen's Printer for Ontario.

Truth and Reconciliation Commission of Canada. (2015). *Calls to action*. Winnipeg, MB: Author. Retrieved from http://www.trc.ca/websites/trcinstitution/File/2015/Findings/Calls_to_Action_English2.pdf

Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences* vol. 366,1881: 3717–25. https://doi.org/10.1098/rsta.2008.0118

Wiseman, D., Lunney Borden, L., Beatty, R., Jao, L., & Carter, E. (2020). STEM Emerging from and contributing to community. *Canadian Journal of Science, Mathematics and Technology Education.*