www.ijemst.net

Facilitating Mathematics and Computer Science Connections: A Cross-Curricular Approach

Kimberly Evagelatos Beck
Utah State University, United States

Jessica F. Shumway
Utah State University, United States

Umar Shehzad
Utah State University, United States

Jody Clarke-Midura
Utah State University, United States

Mimi Recker
Utah State University, United States

To cite this article:

Beck, K.E., Shumway, J.F., Shehzad, U., Clarke-Midura, J., & Recker, M. (2024). Facilitating mathematics and computer science connections: A cross-curricular approach. *International Journal of Education in Mathematics, Science, and Technology (IJEMST), 12*(1), 85-98. https://doi.org/10.46328/ijemst.3104

International Journal of Education in Mathematics, Science and Technology

2024, Vol. 12, No. 1, 85-98 https://doi.org/10.46328/ijemst.3104

Facilitating Mathematics and Computer Science Connections: A Cross-Curricular Approach

Kimberly Evagelatos Beck, Jessica F. Shumway, Umar Shehzad, Jody Clarke-Midura, Mimi Recker

Article Info

Article History

Received:
09 December 2022
Accepted:
07 June 2023

Keywords

Mathematics education
Computer science education
Interdisciplinary curricula
Elementary education

Abstract

In the United States, school curricula are often created and taught with distinct boundaries between disciplines. This division between curricular areas may serve as a hindrance to students' long-term learning and their ability to generalize. In contrast, cross-curricular pedagogy provides a way for students to think beyond the classroom walls and make important connections across disciplines. The purpose of this paper is a theoretical reflection on our use of Expansive Framing in our design of lessons across learning environments within the school. We provide a narrative account of our early work in using this theoretical framework to co-plan and enact interdisciplinary mathematics and computer science (CS) tasks with a team of elementary school educators and school district personnel. The unit focuses on the concepts of exponents in mathematics and repeat loops as a control structure in computer science. Using a narrative approach, we describe what occurred during the collaborative planning of lessons and subsequent enactments in two fifth-grade classrooms and one computer lab and provide a practitioner-oriented account of our experience.

Introduction

Traditional school curricula often exist in separate, distinct silos in the United States. These clear-cut boundaries between curricular disciplines may hinder students' long-term learning and limit generalizations (Engle et al., 2012), ultimately leading learners to erroneously believe that writing is just for English class, art is only created in the studio, computer science exists solely in the computer lab, or that mathematics only applies in the mathematics classroom. Instead, teaching in a way that builds cross-curricular connections encourages students to think beyond the classroom walls and make important connections that can support ownership of and agency in their learning.

However, designing and enacting interdisciplinary lessons in schools presents challenges. One of the difficulties of effective cross-curricular pedagogy is that collaborative planning (co-planning) and enactment of interdisciplinary lessons is a highly complex process. It requires educators and administrators to work closely together to identify concepts that span disciplines, design multiple appropriate classroom activities, and coordinate complicated schedules to enact the lessons.

The purpose of this paper is a theoretical reflection on our use of Expansive Framing in our design of lessons across learning environments within the school. We provide a narrative account of our early work in using this theoretical framework to co-plan and enact interdisciplinary mathematics and computer science (CS) tasks across two learning spaces in elementary school, namely the mathematics classroom taught by an elementary teacher and the computer lab classroom taught by a paraprofessional computer lab specialist (CLS). In this article, we first situate our work in the literature and describe our theoretical lens (Expansive Framing) as a framework for designing cross-curricular, cross-classroom goals for learning. We then describe the Research-Practice Partnership (Penuel et al., 2011, 2020) that structured this collaboration. Next, we outline our experiences in planning and enacting fifth-grade cross-curricular tasks on the topics of exponents (mathematics) and repeat loops (computer science) as framed by the literature and theory of Expansive Framing, followed by vignettes from the mathematics classroom and the computer lab. We conclude with a discussion about the teachers' and CLSs' perspectives on the tasks and connections across the classroom spaces.

Research on Mathematics and Computer Science Connections in Elementary School

Much of the research literature on elementary computer science is based on studies on computational thinking (CT) and coding. Shute et al. (2017) define CT as "the conceptual foundation required to solve problems effectively and efficiently…with solutions that are reusable in different contexts" (p. 151). One way to experience CT is through coding, and researchers argue that CT and coding need to be taught within science and mathematics contexts (Weintrop et al., 2016). Hence, there is a growing body of literature on the integration of CT within mathematics teaching in elementary school. These studies range from analyses of synergies across content domains and standards (Pérez, 2018; Rich et al., 2019) and teachers' planning of integrated math-coding units (Israel & Lash, 2020) to studies of students' mathematics learning embedded in programming environments (Rodríguez-Martínez et al., 2019) and students' use of programming as a virtual manipulative (Goldenberg et al., 2021). Our research is situated in this body of research on connections between mathematics and CT, and our work focuses on computer science instruction in elementary school. Because our research takes place within the traditional siloed structure of the school environment, we use the work on the integration of CS in mathematics and extend it by emphasizing cross-curricular connections across classroom spaces (e.g., the math class and the computer lab) in our lesson design approach.

Expansive Framing

We use Expansive Framing as a theoretical approach to examine how learning mathematics and CS content can be transferred across settings and contexts. We acknowledge that the notion of how and whether transfer of learning happens has long been debated (e.g., Barnett & Ceci, 2002; National Research Council, 2000). Expansive Framing re-conceptualizes transfer through a situative and sociocultural lens, positing that content knowledge is tied with a context of use, and the way a context is framed affects how the content knowledge is used elsewhere. Through this lens, transfer can be promoted by framing topics broadly: across time, place, roles, groups of people, and topics (Engle et al., 2012). Teaching content expansively encourages learners to create expectations that content learned in one setting can be used in other settings, to draw upon relevant prior knowledge, and promote authorship in their own learning. The opposite of expansive framing is bounded framing, which views learning as a one-time occurrence with no application outside of a single event. School curricula are often framed in a bounded manner, yet Engle et al. argue that this method of bounded framing discourages students' long-term learning.

Though mathematics concepts are abundant in CS and coding (Hickmott et al., 2018; Papert, 1980; Rich et al., 2019; Shumway et al., 2021; Weintrop et al., 2016), the relative scarcity of interdisciplinary curriculum that facilitates student discovery of these connections further deepens the divide between disciplines. Our curriculum unit aims to give students repeated exposure to a difficult fifth-grade mathematics topic (in this example, the topic of exponents) and illustrate how the same concepts are used and applied in computer science. Through this partial integration (Nordby et al., 2022), learning is facilitated in and across different contexts by leveraging computational thinking in the computer lab to enhance student understanding of mathematics. Breaking down curricular silos and framing mathematics within the CS environment may help students understand mathematics more deeply and apply it outside of the classroom more effectively (Shehzad et al., 2023).

Research-Practice Partnership: Defining a Problem of Practice

This project is situated within the context of a research-practice partnership (RPP). An RPP is an approach to solving educational problems in which a research organization engages in a long-term collaboration with a practice organization to address the practice partner's educational problem, called a "problem of practice" (Penuel et al., 2011). In this case, the research organization is a university, and the practice organization is a rural-serving school district in the western United States. Thus, unlike traditional approaches where research organizations define and propose solutions to educational problems, the RPP approach is driven by the practice partner's needs and their identified problem of practice. The RPP then works jointly to propose, test, and revise solutions to their problem (Penuel et al., 2020). The RPP approach is claimed to yield more actionable, effective, and sustainable solutions for improving teaching and learning outcomes (Coburn & Penuel, 2016).

In our RPP project, the problem of practice centered around the need to provide high-quality, equitable CS instruction at the elementary school level. The school district context, like many other public school districts, has limited resources and instructional time, with few elementary educators who have the requisite backgrounds to teach CS. In this district, CS is taught by a paraprofessional CLS as an additional course outside of the regular classroom. Students attend their computer lab only once per week for roughly 45 minutes of instruction, which includes both keyboarding and programming skills. Fifth graders in this district have significantly more time devoted to mathematics instruction: around one hour per day, taught by their regular classroom teacher.

To address the problem of practice, project leaders assembled a design team, consisting of three CLSs, two fifth-grade teachers, two District Curriculum Specialists, and university researchers. The proposed solution was to integrate CS into mathematics instruction, in particular to introduce CS concepts *within* the mandated fifth-grade mathematics curriculum (*GO Math!* by Houghton Mifflin Harcourt Publishing) during classroom instruction.

Students would then further explore these ideas in their programming lessons during weekly computer lab time taught by paraprofessional CLSs. District Curriculum Specialists requested that the design team focus on programming lessons in Scratch (scratch.mit.edu), the block-based programming platform they were planning to use with the fifth-grade students. The design team collaboratively planned several integrated mathematics and CS tasks that supplemented regular instruction and that spanned the computer lab and classroom settings.

Research-Practice Partnership: Defining a Problem of Practice

In the first meeting, the design team collaboratively identified topics in the existing mathematics curriculum that are traditionally difficult for fifth-grade students. Teachers pinpointed exponents as a concept that the existing curriculum did not address sufficiently, leaving students with a limited view of the topic's applicability. Teachers noted that their students also struggled to differentiate exponents from multiplication, observing that they frequently misinterpreted an exponent as multiplication and made conceptual errors such as $3^2=6$. We discussed the need for students to understand the difference between repeated addition of addends (multiplication) and repeated multiplication of factors (exponents). Discussions then turned to ideas about how exponents could be connected to ideas in programming. The concept of repeat loops was discussed as a possible way to teach the repetition of patterns in exponents.

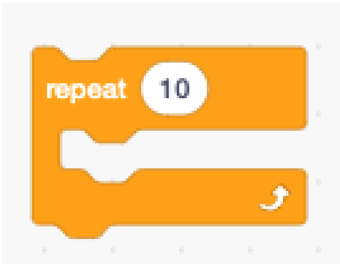

Figure 1. Repeat Loop Block in Scratch

Repeat loops are used in computer programming to iterate a section of code a given number of times. The repeat loop block in Figure 1 is a commonly used control structure in Scratch programming. Using Expansive Framing as a lens, the design team drew parallels between this computer programming concept of repeat loops and the mathematical concept of exponents as repeated multiplication (see Figure 2).

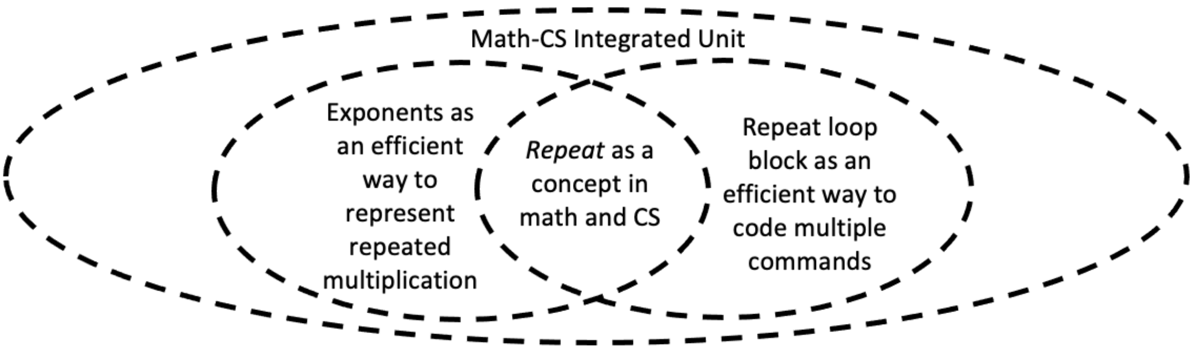

Figure 2. Expansive Framing: An Example of Connecting Math and CS Concepts

The design team created four tasks to be enacted in the mathematics classroom (two adapted from existing curriculum and two newly created) and five tasks to be enacted in the computer lab over the course of two class sessions. Table 1 shows both sets of tasks in sequential order. The tasks were designed to supplement the existing instruction, either as a warm-up activity or as an extension to existing activities.

Table 1. Mathematics and Computer Science Lesson Names and Sequencing

Lesson Name	Classroom Activity	Sequence
Math Class Tasks #1-2	#1: What Makes This Equation True? (Learning objective: **Repeated addition and repeated factors** to supplement *GO Math!* lessons) #2: Is This Equation True or False? (Learning objective: **Repeated addition and repeated factors** to supplement *GO Math!* lessons)	Teach prior to Computer Lab tasks
Computer Lab Tasks #1-5	#1: **Repeated Addition and Repeated Multiplication** #2: Visualizing Growth by **Multiplication** #3: Visualizing **Exponential** Growth #4: Comparison of Growth by **Multiplication and Exponents** #5: Writing Your Own Code	Teach during the typical Computer Lab lessons on looping and teach over two class sessions
Math Class Tasks #3-4	#3: Visualizing **Repeating Factors** (using problems from *GO Math!* with the Scratch programming visualizations) #4: **Efficient Exponents** (using problems from *GO Math!* with the Scratch programming visualizations)	Teach after Computer Lab tasks

Note. Connections between the mathematics classroom and the computer lab are shown in bold.

The first two mathematics tasks were 15-minute number strings, a common warm-up or math talk routine, on the foundations of exponents. The purpose of these tasks for the mathematics classroom was to prime students' prerequisite conceptual knowledge of differences between repeated addition and repeated multiplication for the computer lab. In these activities, students explored the idea of repeated addition and repeated multiplication by discussing prompts such as how to make the equation 4 + 4 + 4 + 4 + __ + __ = 6 x 4 true, and whether 4 x 3 = 4 x 4 x 4 is true or false and why.

Following the two math tasks, the students participated in a series of computer lab tasks focusing on repeat loop blocks. The design team developed the computer lab tasks based on the Use-Modify-Create framework (Lee et al., 2011), which scaffolds computational thinking by first having students run (Use) an existing program, followed by revising and refining others' programs (Modify), and finally by generating their own unique programs (Create). Since these tasks were taught toward the beginning of the school year and the fifth graders had only a little prior experience with Scratch, we primarily designed activities at the "Use" end of the continuum. Students were provided with pre-programmed Scratch shells that contained snippets of code and tasked with assembling and testing basic programs. Mathematics content provided the context of these programs (for example, coding an animated dinosaur to perform repeated addition and predicting the result), and repeat loop blocks played a prominent role.

Finally, the fifth graders revisited the mathematical and computer programming concepts during math class tasks #3 and #4. These tasks adapted content from the district's adopted *GO Math!* textbook and paired it with programming concepts the students had learned in the computer lab. For example, students were asked to use individual white boards to write 4^3 in word form ("four to the third power"), exponent form (4^3), and expanded form (4 x 4 x 4), then watch a visualization of exponential growth programmed in Scratch (Figure 3). In Figure 3, the Scratch code for 4^3 is shown on the left. When the code is executed (by clicking the green flag), the program presents a visualization of 4^3, ending with the image on the right. In this example, the number of orange cats drawn on the screen grew exponentially based on the values given to the variables *base* and *exponent* as the program's input.

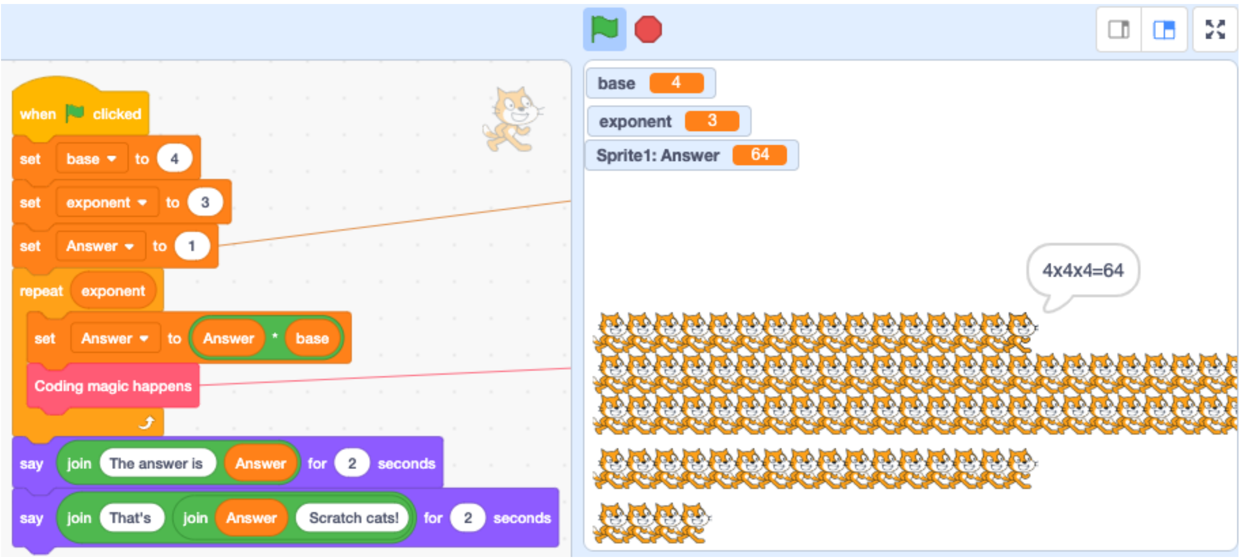

Figure 3. Scratch Visualization of Repeated Multiplication

As Figure 3 shows, repeat loop blocks were a key feature of the program as a means to reintroduce the mathematical concept of exponents as repeated multiplication and represent the concept in a dynamic visualization. Because this was early in the year and students had not yet learned Scratch, the district requested the creation of a "Coding magic happens" block to abstract the details of the long procedures used in this particular program. In current lessons with students familiar with Scratch, we have replaced the "Coding magic happens" block with instructions to create multiplication and exponent operations using procedures (My Blocks) in Scratch code.

In another activity, fifth graders explored the difference between 2 x 5 and 2^5, an example of a concept that teachers identified as causing confusion for their students. First, students discussed their hypotheses for how 2 x 5 was different than 2^5 in small groups, then as a class they viewed another visualization to demonstrate the difference between the two operations (see Figure 4). Figure 4 depicts a Scratch program, based on the code shown in Figure 3, that created visualizations of multiplicative growth and exponential growth side by side to highlight the difference between multiplication (orange cats) and exponents (gray cats). The code used in the activity took two numbers as input (*first number* and *second number*). The two numbers were then used in two different calculations. In one calculation, *first number* and *second number* were used in a multiplication operation. In the other calculation, *first number* was used as the base and *second number* as the exponent. The calculations produced two visualizations in the output window, side by side (see Figure 4).

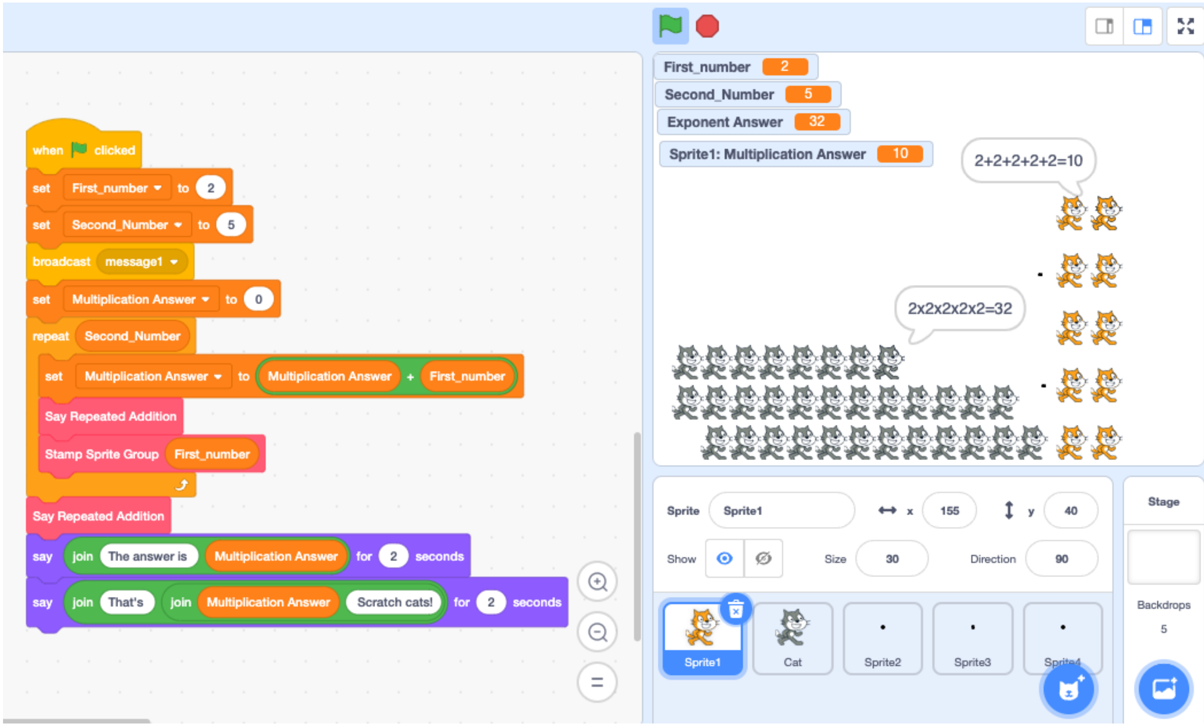

Figure 4. Scratch Visualization of Repeated Addition Versus Repeated Multiplication

Note. Scratch code for repeated addition is shown on the left-hand side. See figure 3 for code for repeated multiplication

Figures 3 and 4 show how the computer science concept of repeat loops is used to help students understand the mathematical concepts of repeated addition and repeated multiplication more deeply. The visualizations produced from the code allowed students to see a visual representation of the concept through the stamping of the cats (the answer) and the equations written on the screen. By juxtaposing the two concepts in Figure 4, the students were able to see the visual differences between them and that the number of gray cats stamped on the screen grew much faster than the number of orange cats. Students learned about the power of programming and CS as tools for making concepts more concrete through visualizations while also learning about the concepts more deeply.

The overall unit highlights the idea of efficiency of representation through repetition, an anchor concept that we named "repeats." Multiplication, which is already familiar to the fifth graders, is reinforced as an efficient way to represent repeated addition. This leads to the idea that exponents are an efficient way to represent repeated multiplication, and the repeat loop block is an efficient way to represent this idea in code through abstraction.

Drawing upon our theoretical framework, we drew parallels between programming and mathematics in each task. Students in the mathematics classroom used what they learned in the computer lab about repeat loops to reinforce their understanding of exponents as repeated multiplication. Conversely, students in the computer lab reinforced their understanding of repeat loops by drawing upon their knowledge of repeated operation of multiplication to author their program. Thus, our design goal was to expansively frame exponents and repeat loop blocks, tied together by a common thread of efficiency by repetition, in order to promote student transfer and deeper long-term learning of both mathematics and computer science principles.

Task Enactment in the Mathematics Classroom and Computer Lab

The full sequence of tasks (both mathematics and computer science) was enacted in one rural elementary school. Students in the school's two-fifth grade classes (approximately 30 students each) learned the mathematics during their regular classroom instruction, taught by Mrs. A and Mrs. W. They then learned the coding concepts during their regular computer lab time, taught by Mrs. J. In the following sections, we take a narrative approach to describe notable cross-curricular connections during task enactment in both the computer lab and the mathematics classroom.

Computer Lab Tasks on Repeat Loops: Expansively Framing Exponents

In the following vignette, Mrs. J, a paraprofessional CLS, showed fifth-grade students the Scratch visualization program for Task #3: Visualizing Exponential Growth (see Figure 3) that took two variables as input: one called the *base* and the other called the *exponent*:

> *Mrs. J*: …this one's kind of the cool one. You can see that this code we are setting base to four. And … our exponent is three. What's another way to say that? So, our base is four and our exponent is three.
> *Zac*: Four to the third power.
> *Mrs. J*: Nice, four to the third power. The value of the exponent is how many times the base is multiplied.
> *Mrs. J*: …in this example, exponent is three inside your repeat loop…our base is four, our exponent is three. How many cats when I click on the green flag? How many cats do you think will appear? Amanda?
> *Amanda*: 64

In this interaction, Mrs. J intentionally used the terms *base* and *exponent* and explained an exponent's mathematical function in order to establish the mathematical context and make connections to the language used in the mathematics classroom. Mrs. J then referred to the *exponent* variable in the code and to the mathematics inside the program. Mrs. J clicked the green flag, which ran the program, and the output produced a visualization of the exponential growth of 64 Scratch cats. Here, the students encountered the math concept of exponents in the context of learning about the repeat block in the computer program. This is one example of how cross-curricular connections provided students an opportunity to interact with exponents and exponential visuals in another space (the computer lab). Expansively framed tasks, vocabulary and visuals are intended for deeper learning of concepts. In this example, Mrs. J stops short of having Amanda explain how she knows and what that might look like when they run the program, which could lead into a deeper discussion of "repeats" across mathematics and CS. Hence, there is a need for planned teacher questions that can elicit the expansively framed concept of "repeat."

The following example comes from a subsequent lab session in Mrs. J's computer lab in which she introduced Task #4: Comparison of Growth by Multiplication and Exponents to students. Referring to the *first number* and *second number* variables in Figure 4, Mrs. J asked what those numbers mean in the code and what students expected to see in the output:

> *Mrs. J*: Our numbers up here look a little bit different [from the other program]…It's doing two different things. It's multiplying…so we're going to say this is two times five. And then what else is it also saying? Two to the?
>
> *Multiple students*: Power of five.
>
> *Mrs. J*: Two to the power of five or two to the fifth power...Let's look at this. What do you [think will] happen?
>
> *Alice*: So, this one on the gray cat is going to have two as the base and five as the exponent, okay, and then it's going to make like groups of that number. And then on the orange cat is going to have, groups of like two or five.

Alice was able to read the code and attempted to describe how the two numbers would be represented in the program's output. Mrs. J then ran the program, which corresponded with portions of Alice's description of the output. Recall that we used "coding magic" to abstract from some of the more complex coding procedures. Despite this abstraction, this student was able to read and decipher the code while using her emerging knowledge of exponential versus multiplication outputs. Expansive Framing would suggest that Alice will then be more likely to transfer this knowledge of the numbers in the code to better use and understand base, exponent, and groups in mathematics class.

Mathematics Tasks on Efficiency of Exponents: Expansively Framing Repeat Loops

This section focuses primarily on fifth-grade teacher Mrs. A's enactment of Math Task #4: Efficient Exponents in the mathematics classroom. This task focused on the conceptual differences between multiplication (repeated addition) and exponents (repeated multiplication) and took place after the students had participated in the computer lab lessons. This task revisited the concepts of exponents and the repeat block and was designed to facilitate connections across mathematics and CS.

> *Mrs. A:* Think about the time that you spent in the computer lab recently. And tell me what have you learned about repeating in your coding?
>
> *Simon:* You just repeat everything over and over again.
>
> *Mrs. A:* Okay, you can just repeat things over and over. What do you remember, Rose?
>
> *Rose:* Say you only can have eight blocks. And if you do move forwards and stuff, it would add up to be 11 blocks. So, you can do a repeat.
>
> *Mrs. A:* Oh, so you're saying it's kind of a shortcut? It's a shortcut to what? To do a repeated function, right?

Here Mrs. A noted how Rose identified repeats as a shortcut ("Say you can only have eight blocks…") and helped her students activate their background knowledge of the programming concept of repeat loops to represent code segments more efficiently. This representation helps students abstract away from the details of individual code segments and instead focus on the concepts relevant to understanding. Mrs. A then expansively framed the notion of repeats by including a mathematics activity. Students viewed a program from Computer Lab Task #4, Comparison of Growth by Multiplication and Exponents to clarify the distinction between repeated addition (6 + 6 + 6) and repeated multiplication (6 x 6 x 6). Some students had trouble explaining the distinction prior to running the program that produced the visualizations. To further explore the idea, Mrs. A posed a related question to her class:

> *Mrs. A:* Let's look at the three to the power of two. What does this mean? Let's start, three to the power of two, which is…?
>
> *Samson:* Six. No…nine?
>
> *Mrs. A:* Yeah, nine, you were adding it.

Here Samson displayed the common conceptual error that the fifth-grade teachers anticipated their students would make when learning about exponents: confusing repeated addition (3 + 3) with repeated multiplication (3 x 3). Mrs. A then reminded the students of the difference between repeated addition and repeated multiplication, and the class worked together to calculate $3^2 = 9$.

> *Mrs. A:* What did you notice?
>
> *Kate:* They get a lot bigger, faster than multiplication.
>
> *Mrs. A:* Faster than multiplication…I like what you're saying because multiplication is repeated addition, right? And here we were doing repeated multiplication.

In her statement, "They get a lot bigger, faster than multiplication," it is not clear if Kate is comparing the outputs only or if she understands that repeated multiplication (exponents) yields much faster growth than repeated addition (multiplication). Nevertheless, Kate's statement shows an emerging idea of exponential growth. Next, students worked with a partner to come up with multiple representations of the expression 6^3 on a whiteboard. Mrs. A asked her class to explain what they had discussed. Some students struggled to interpret 6^3:

> *Kate:* Six times six, six times?
>
> *Carmella:* Three times three times three, six times.
>
> *Samson:* Six plus six plus six plus six, like three rows of six plus six, six times.

Based on these responses, it appears that at this point in the lesson, Kate, Carmella, and Samson were not able to explain the meaning of an exponential expression with six as the base and three as the exponent. Mrs. A continued the class discussion, eventually guiding students to the conclusion that 6x3 and 6^3 are similar because they both repeat the six, but are different in the operation that is used (repeated addition versus repeated multiplication). Mrs. A projected the Scratch visualization from Computer Lab Task #4 (see Table 1), Comparison of Growth by Multiplication and Exponents, to show repeated addition versus repeated multiplication (see Figure 4) using two as the base and five as the exponent. The class reacted enthusiastically, and students were able to explain their thinking more clearly.

Zada: Oh, I see it now!

Rose: Oh, that makes more sense.

Gavin: Okay. So, the orange cats did five times two. Okay, to get 10. And then the gray cats did five to the power of two to get 32? Yeah. Two to the fifth. I'm smart, guys.

Gavin accurately described the difference between the growth of the orange and gray cats and expressed confidence in his understanding of the mathematics illustrated through this Scratch program. While the pre-visualization discussion showed some students' fragile knowledge of the meaning of an exponential expression, the Scratch visualization led to more buzz ("Oh, I see it now!") and better explanations of the concept of exponents, making an abstract concept more concrete through the power of programming. This highlights the power of integration of coding in mathematics, and the theory of Expansive Framing would suggest that Gavin will transfer in a deeper understanding of repeat loops to the computer lab lessons because he used repeat loops and their output in mathematics class to visually understand exponential growth.

Cross-Curricular Connections

The Design Team used Expansive Framing as a theoretical lens to develop computer lab lessons for Mrs. J's class and mathematics lessons for Mrs. A's class. In these lessons, expansively-framed design features included 1) teacher questions that highlight connections and the anchor idea of repeats (e.g., "...what have you learned about repeating in your coding?") and 2) common use of visualizations of outputs. In Mrs. A's class, she used the visualizations from their computer lab Scratch lessons to "see" outputs to problems in their mathematics textbook. Mrs. J and Mrs. A capitalized on these design features and used the Scratch visuals and connections questions to assist students in making cross-curricular associations. Expansive Framing theory explains that the enactment of these design features makes it more likely for students to use the exponential growth concept outside of math class, and in particular, recognize and use it more readily in the computer lab. Similarly, the enactment of the design features may make it more likely for students to have a deeper understanding of the repeat loop in coding and use this concept in mathematics class.

Teachers' and CLSs' Perspectives on Enacting Expansively Framed Tasks

After classroom implementations, the design team met to discuss what went well, areas of difficulty, what was learned, and what to change for future cross-curricular lessons. The mathematics teachers reported that their students displayed higher-level thinking about exponents and repeating patterns in mathematics. For example, some students wondered aloud whether $3^6 = 6^3$ and were able to reason through it mathematically and test their conjectures using the Scratch program. The teachers also felt that the recurring exposure to the concept of exponents in multiple settings was ultimately beneficial to their students, but we also wonder if it could have been the dynamic visualizations in Scratch that added "salience and interest" (Goldenberg et al. 2021, p. 56). By the time the unit wrapped up, the teachers reported that most students had a solid grasp on the notion of exponents. The team unanimously agreed that some of the most effective features of the lessons were the Scratch programs (see Figures 3 and 4) that allowed the children to visualize the difference between multiplication as repeated addition and exponents as repeated multiplication. The fifth graders became more animated and excited when they were able to play with the program and test different numbers. Many of them entered progressively larger numbers to try and stump the computer, all while consolidating their newly learned concept of exponents.

The mathematics teachers expressed that the challenges to this cross-curricular unit were largely logistical, such as difficulty coordinating mathematics classroom and computer lab schedules to ensure the proper sequence and tasks taking too much time. The CLSs and mathematics teachers noticed students' difficulty in remembering mathematics vocabulary when in the computer lab, even terms that they use regularly such as "factor." This led to a design team discussion about finding better ways to activate cross-curricular background knowledge at the beginning of each lesson by providing definitions of pertinent terminology across disciplines and starting each lesson activating students' descriptions of what they had learned in the other classroom. For example, in future lessons, we plan to add educative (Davis & Krajcik, 2005) mathematics elements to the computer lab lessons to help students make stronger connections between mathematics and coding, which we think will also help CLSs to better facilitate some discussion among students about the cross-curricular connections.

Conclusions

The purpose of this article was to report our RPP project's cross-curricular approach to facilitate mathematics and CS connections across the mathematics classroom and computer lab through an expansively framed unit on exponents and the concept of repetition and efficiency in representation in programming and mathematics. The partner district in this project identified a need to provide CS instruction to all fifth-grade students through computer lab instruction by paraprofessional CLSs and the RPP proposed a solution through the integration of CS into mathematics instruction. Rather than approach these classrooms separately, we worked on an interdisciplinary unit to provide students with repeated opportunities to learn a challenging mathematics topic (exponents) and understand the concept within the CS context. The aim of this unit was to facilitate learning and connections across classroom spaces, breaking down the silos of mathematics and CS in elementary school.

The narrative in this paper illustrates our experience in co-planning and enacting an expansively framed curriculum. Our aim was to provide a practitioner-oriented account of what occurred in the initial stages of this process. There are some inherent limitations in this study. This narrative account provided a very fine-grained look at what happened in one mathematics classroom and one computer lab. Future research should take a broader scope and investigate connections across multiple classrooms.

Further, there are limitations in what can be inferred from students' and teachers' statements in the present study. Future work should formulate research questions and employ corresponding instruments and tools to identify and measure connections that are made across contexts. At this point, the design team has taken the lessons learned from our first design iteration to continue creating units that connect across mathematics and CS within this RPP. Our next steps are to more deeply investigate how the expansive framing of mathematics and CS concepts influences students in making connections across classroom spaces and disciplinary concepts in ways that enhance their learning and engagement.

Acknowledgement

We thank the participating teachers for their generous time and involvement in this work. This work was supported by funding from the National Science Foundation under Grant Nos. 2031382 and 2031404. The opinions expressed herein are those of the authors and do not necessarily reflect those of the National Science Foundation.

References

Barnett, S. M., & Ceci, S. J. (2002). When and where do we apply what we learn?: A taxonomy for far transfer. *Psychological Bulletin*, *128*(4), 612–637. https://doi.org/10.1037/0033-2909.128.4.612

Coburn, C. E., & Penuel, W. R. (2016). Research-practice partnerships in education: Outcomes, dynamics, and open questions. *Educational Researcher, 45*(1), 48–54. https://doi.org/10.3102/0013189X16631750

Davis, E. A., & Krajcik, J. S. (2005). Designing educative curriculum materials to promote teacher learning. *Educational Researcher*, *34*(3), 3–14. https://doi.org/10.3102/0013189X034003003

Engle, R. A., Lam, D. P., Meyer, X. S., & Nix, S. E. (2012). How does expansive framing promote transfer? Several proposed explanations and a research agenda for investigating them. *Educational Psychologist, 47*(3), 215–231. https://doi.org/10.1080/00461520.2012.695678

Goldenberg, E. P., Carter, C. J., Mark, J., Reed, K., Spencer, D., & Coleman, K. (2021). Programming as language and manipulative for second-grade mathematics. *Digital Experiences in Mathematics Education, 7*, 48-65.

Hickmott, D., Prieto-Rodriguez, E. & Holmes, K. (2018). A scoping review of studies on computational thinking in K-12 mathematics classrooms. *Digital Experiences in Mathematics Education, 4,* 48-69.

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

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., Malyn-Smith, J., & Werner, L. (2011). Computational thinking for youth in practice. *ACM Inroads, 2*(1), 32–37.

National Research Council (2000). *How people learn: Brain, mind, experience, and school* (expanded ed.). The National Academies Press. https://doi.org/10.17226/9853

Nordby, S.K., Bjerke, A.H. & Mifsud, L. (2022). Computational thinking in the primary mathematics classroom: A systematic review. *Digital Experiences in Mathematics Education, 8*, 27–49.

Papert, S. A. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic books.

Penuel, W. R., Fishman, B. J., Haugan Cheng, B., & Sabelli, N. (2011). Organizing research and development at the intersection of learning, implementation, and design. *Educational Researcher*, *40*(7), 331–337.

Penuel, W. R., Riedy, R., Barber, M. S., Peurach, D. J., LeBouef, W. A., & Clark, T. (2020). Principles of collaborative education research with stakeholders: Toward requirements for a new research and development infrastructure. *Review of Educational Research, 90*(5), 627–674.

Pérez, A. (2018). A framework for computational thinking dispositions in mathematics education. *Journal for Research in Mathematics Education, 49*(4), 424-461.

Rich, K. M., Spaepen, E., Strickland, C., & Moran, C. (2019). Synergies and differences in mathematical and computational thinking: Implications for integrated instruction. *Interactive Learning Environments,* *28*(3), 272–283.

Rodríguez-Martínez, J. A., González-Calero, J. A., & Sáez-López, J. M. (2019). 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

Shehzad, U., Clarke-Midura, J., Beck, K., Shumway, J., & Recker, M. (2023). Co-designing elementary-level computer science and mathematics lessons: An Expansive Framing approach. In *Proceedings of the International Conference of the Learning Sciences* (pp. 902-905). Montreal, Canada: International Society of the Learning Sciences.

Shumway, J. F., Welch, L., Kozlowski, J., Clarke-Midura, J., & Lee, V. R. (2021). Kindergarten students' mathematics knowledge at work: The mathematics for programming robot toys. *Mathematical Thinking and Learning.* Online first. https://doi.org/10.1080/10986065.2021.1982666

Shute, V. J., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review*, *22*, 142-158.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147.

Author Information

Kimberly Evagelatos Beck
https://orcid.org/0000-0001-6936-6607
Utah State University
United States
Contact e-mail: *kimberly.beck@usu.edu*

Jessica F. Shumway
https://orcid.org/0000-0001-7655-565X
Utah State University
United States

Umar Shehzad
https://orcid.org/0000-0002-7922-4132
Utah State University
United States

Jody Clarke-Midura
https://orcid.org/0000-0001-5434-0324
Utah State University
United States

Mimi Recker
https://orcid.org/0000-0002-5372-4403
Utah State University
United States