

Utah State University

DigitalCommons@USU

---

Publications

Integrating Elementary-Level Mathematics  
Curricula with Expansively-Framed Computer  
Science Instruction

---

4-2023

## Geometry and Coding: Introducing an Interactive and Integrated Mathematics-Computer Science Unit

Kimberly Beck

*Utah State University*, [kimberly.beck@usu.edu](mailto:kimberly.beck@usu.edu)

Jessica F. Shumway

*Utah State University*, [jessica.shumway@usu.edu](mailto:jessica.shumway@usu.edu)

Follow this and additional works at: [https://digitalcommons.usu.edu/eled\\_support\\_pubs](https://digitalcommons.usu.edu/eled_support_pubs)



Part of the [Computer Sciences Commons](#), [Curriculum and Instruction Commons](#), [Elementary Education Commons](#), and the [Mathematics Commons](#)

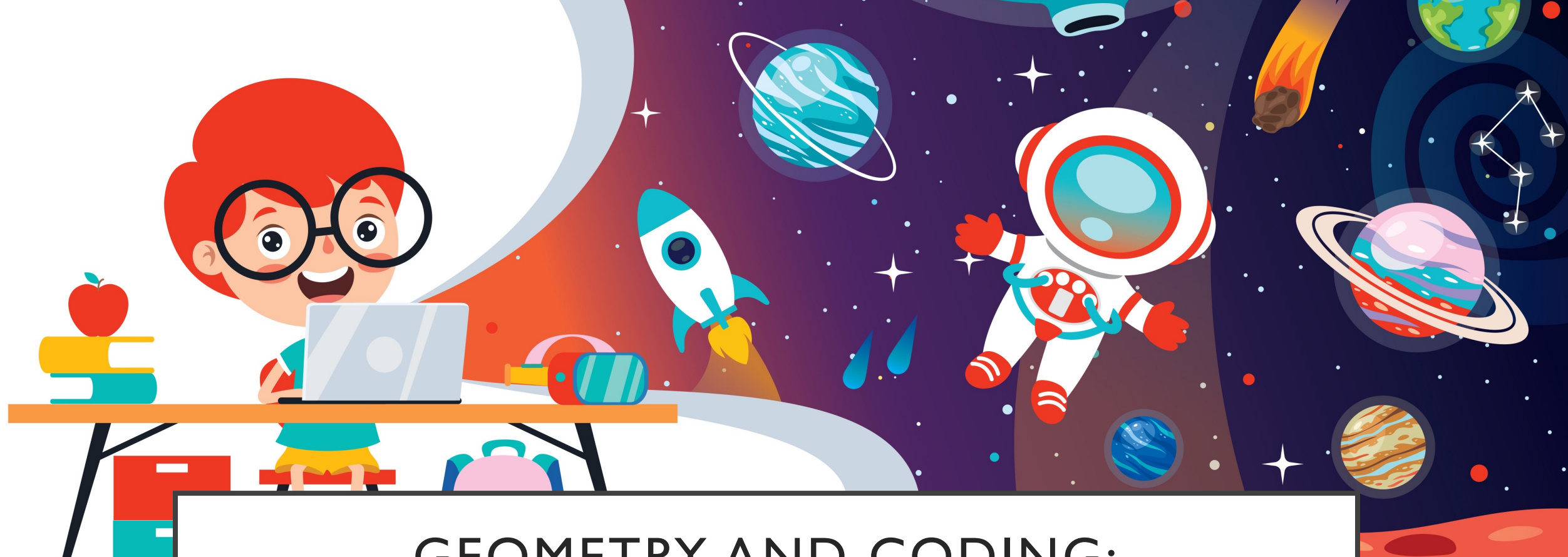
---

### Recommended Citation

Beck, K., & Shumway, J. F. (2023). Geometry and Coding: Introducing an Interactive and Integrated Mathematics-Computer Science Unit. Presentation at the National Council of Teachers of Mathematics 5. [https://digitalcommons.usu.edu/eled\\_support\\_pubs/9](https://digitalcommons.usu.edu/eled_support_pubs/9)

This Curriculum is brought to you for free and open access by the Integrating Elementary-Level Mathematics Curricula with Expansively-Framed Computer Science Instruction at DigitalCommons@USU. It has been accepted for inclusion in Publications by an authorized administrator of DigitalCommons@USU. For more information, please contact [digitalcommons@usu.edu](mailto:digitalcommons@usu.edu).





**GEOMETRY AND CODING:  
INTRODUCING AN INTERACTIVE AND INTEGRATED  
MATHEMATICS-COMPUTER SCIENCE UNIT**

**Kimberly Beck**

School of Teacher Education  
Utah State University

**Jessica Shumway**

School of Teacher Education  
Utah State University

## SESSION OVERVIEW

1. Cache Code Math Project

2. What is Coding in Elementary School?

3. Expansive Framing

4. Expansively Framing Geometry and Coding

5. Other Areas for Integration and Expansive Framing: Exponents and Fractions

6. Discussion and Q&A



## CACHE CODE MATH



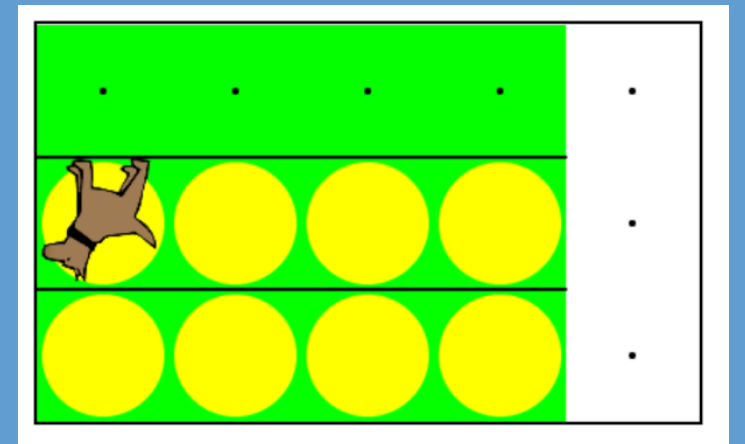
CSFORALL GRANT  
#2031382 &  
#2031404

PIs: Mimi Recker, Jody Clarke-Midura,  
Jessica Shumway, & Victor Lee

# CODING IN ELEMENTARY SCHOOL



# SCRATCH



## WHAT IS SCRATCH?

“Scratch is the world’s largest coding community for children and a coding language with a simple visual interface that allows young people to create digital stories, games, and animations.”

*[scratch.mit.edu/about](https://scratch.mit.edu/about)*



Let's Explore!

1. Go to [scratch.mit.edu](https://scratch.mit.edu)
2. Click on “Start Creating”



SCAN ME



Make a letter turn when you click it.



## GET READY



Go to the  
Sprite Library.



Click the **Letters** category.



Choose a letter sprite.

## ADD THIS CODE



Try different  
numbers.

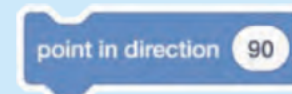
## TRY IT

Click your letter.



## TIP

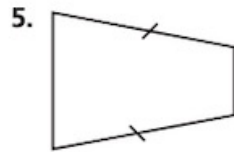
Click this block to reset the sprite's direction.



## Bounded Framing

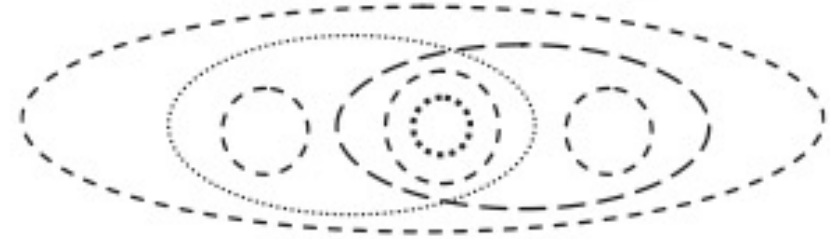


Classify the quadrilateral in as many ways as possible.  
Write *quadrilateral, trapezoid, parallelogram, rectangle, rhombus, or square.*

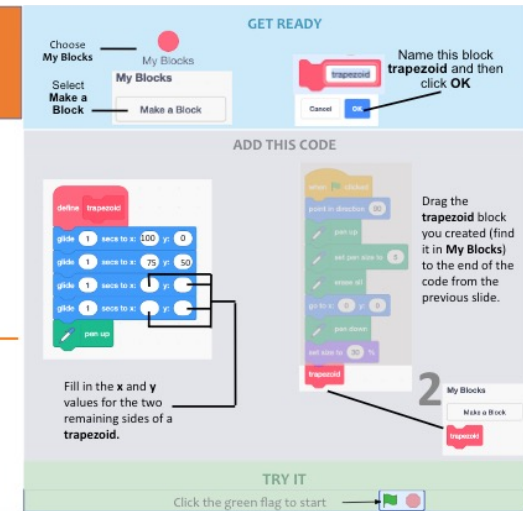
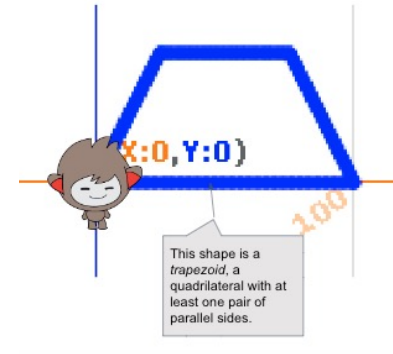


**quadrilateral, trapezoid**

## Expansive Framing



Challenge Task:  
Change the code so that nano draws a trapezoid.



# EXPANSIVE FRAMING

Image credit: 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>



# MATH & COMPUTER SCIENCE STANDARDS

## FCR Focus:

### Common Core State Standards

**5.G.B.3** Understand that attributes belonging to a category of two-dimensional figures also belong to all subcategories of that category.

**5.G.B.4** Classify two-dimensional figures in a hierarchy based on properties.

**MATHEMATICAL PRACTICES** (See *Mathematical Practices in GO Math!* in the *Planning Guide* for full text.)

**MP1** Make sense of problems and persevere in solving them. **MP7** Look for and make use of structure. **MP8** Look for and express regularity in repeated reasoning.

## Algorithms and Programming (AP):

An algorithm is a sequence of steps designed to accomplish a specific task. Algorithms are translated into programs, or code, to provide instructions for computing devices. Algorithms and programming control all computing systems, empowering people to communicate with the world

in new ways and efficiently, breaking it down, analyzing data

Utah K-5 Computer Science Standards

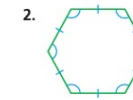
Writing Draft: March 15, 2019

**Standard 4.AP.2 - Create programs that include events, loops, and conditionals.** (*Practice 5: Creating Computational Artifacts*)

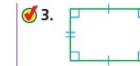
Students will develop a set of instructions (a program) that include events, loops, and conditionals to facilitate and manage tasks. Event examples include mouse clicks, typing on the keyboard, and collisions between objects. Conditional statements are sets of commands that are tied to specific actions based on whether the condition evaluates to TRUE or FALSE. Other terms that can be used to specify the appropriate groups of instructions to execute under various conditions include AND, OR, and NOT.

Name \_\_\_\_\_

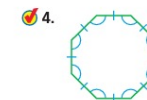
Name each polygon. Then tell whether it is a *regular polygon* or not a *regular polygon*.



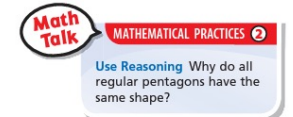
hexagon; regular polygon



quadrilateral; not a regular polygon



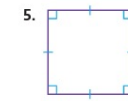
octagon; regular polygon



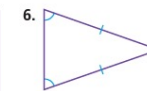
Possible answer: All angles are congruent, so that gives all regular pentagons, no matter how long their sides, the same shape.

### On Your Own

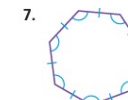
Name each polygon. Then tell whether it is a *regular polygon* or not a *regular polygon*.



quadrilateral; regular polygon



triangle; not a regular polygon



heptagon; regular polygon



hexagon; not a regular polygon

9. **GO DEEPER** Compare the polygons shown in Exercises 2 and 8. Describe how they are alike and how they are different.

Possible answer: Both polygons are hexagons. All sides and angles are congruent in the hexagon shown in Exercise 2, so it is a regular polygon. The hexagon in Exercise 8 is not a regular polygon because the sides have different lengths and not all angles are congruent.

**Common Core State Standards**

**5.G.B.3** Understand that attributes belonging to a category of two-dimensional figures also belong to all subcategories of that category.

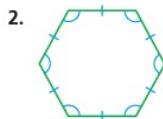
**5.G.B.4** Classify two-dimensional figures in a hierarchy based on properties.

**MATHEMATICAL PRACTICES** (See *Mathematical Practices in GO Math!* in the *Planning Guide* for full text.)

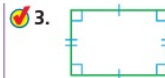
**MP1** Make sense of problems and persevere in solving them. **MP7** Look for and make use of structure. **MP8** Look for and express regularity in repeated reasoning.

Name \_\_\_\_\_

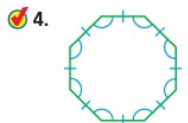
Name each polygon. Then tell whether it is a *regular polygon* or *not a regular polygon*.



hexagon; regular polygon



quadrilateral; not a regular polygon



octagon; regular polygon

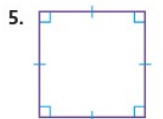
**Math Talk** **MATHEMATICAL PRACTICES 2**

Use Reasoning Why do all regular pentagons have the same shape?

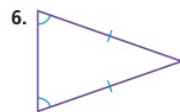
Possible answer: All angles are congruent, so that gives all regular pentagons, no matter how long their sides, the same shape.

**On Your Own**

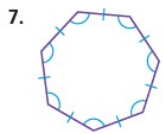
Name each polygon. Then tell whether it is a *regular polygon* or *not a regular polygon*.



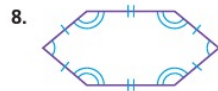
quadrilateral; regular polygon



triangle; not a regular polygon



heptagon; regular polygon



hexagon; not a regular polygon

9. **GO DEEPER** Compare the polygons shown in Exercises 2 and 8. Describe how they are alike and how they are different.

Possible answer: Both polygons are hexagons. All sides and angles are congruent in the hexagon shown in Exercise 2, so it is a regular polygon. The hexagon in Exercise 8 is not a regular polygon because the sides have different lengths and not all angles are congruent.

## Algorithms and Programming (AP):

An algorithm is a sequence of steps designed to accomplish a specific task. Algorithms are translated into programs, or code, to provide instructions for computing devices. Algorithms and programming control all computing systems, empowering people to communicate with the world in new ways and solve compelling problems. The development process to create meaningful and efficient programs involves choosing which information to use and how to process and store it, breaking apart large problems into smaller ones, recombining existing solutions, and analyzing different solutions.

Utah K-5 Computer Science Standards

Writing Draft: March 15, 2019

### Standard 4.AP.2 - **Create programs** that include **events**, **loops**, and **conditionals**. (*Practice 5: Creating Computational Artifacts*)

Students will develop a set of instructions (a program) that include events, loops, and conditionals to facilitate and manage tasks. Event examples include mouse clicks, typing on the keyboard, and collisions between objects. Conditional statements are sets of commands that are tied to specific actions based on whether the condition evaluates to TRUE or FALSE. Other terms that can be used to specify the appropriate groups of instructions to execute under various conditions include AND, OR, and NOT.

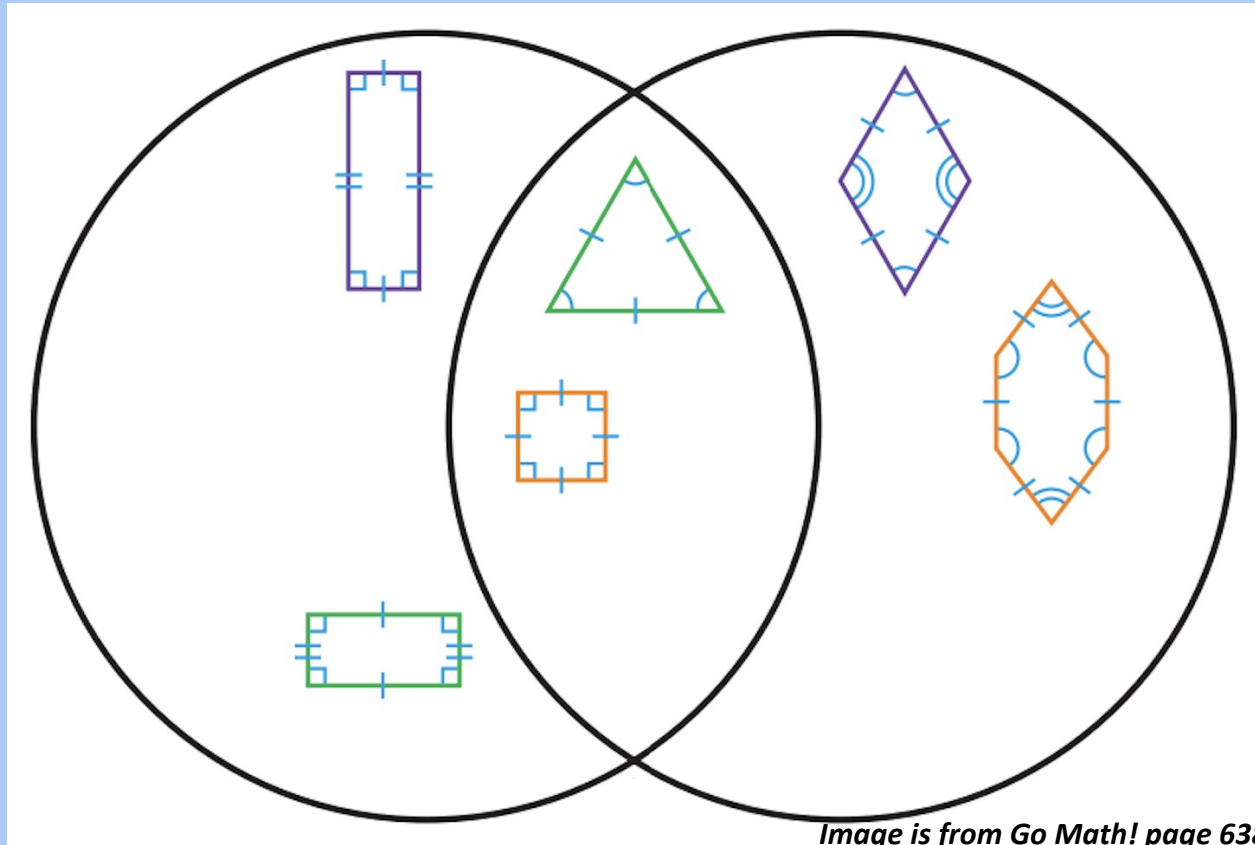


Cache Code Math: CS-Math Unit on Geometry for March  
Chapter 11 (and review Chapter 9, Lesson 9.2) in *Go Math!* + Scratch Cards

Lesson	Computer Lab	Mathematics Class	Notes	Suggested Timing/Pacing
#1 Math Routine		Math Routine: Which One Doesn't Belong?  (reasoning about shapes using attributes in Chapter 11 lessons)	Teach <b>prior to</b> Computer Lab lessons  <i>Supports a math content and language review. Supports reasoning with attributes.</i>	7-10 minutes  Week of March 14, before or during Lesson 11.1
Computer Lab Scratch Card: Introduction to My Block	The typical Scratch Card ( <a href="#">Scratch card 9: Jumping Game Cards</a> )  New jumping card ( <a href="#">Monkey jump</a> )		Teach <b>prior to</b> the Computer Lab Quadrilateral activity.  <i>Supports students' learning of the My Block procedure in Scratch in order to focus on the mathematics in subsequent Computer Lab lessons.</i>	5-10 minutes  Week of March 14  <b>Exit Ticket</b>
#2 Math Minilesson		Math Minilesson: Visualizing the Shape – What Shape Will It Be?  ( <i>extends</i> Lesson 9.2 on order pairs and accesses background knowledge for Lessons 11.1-11.3)	Teach <b>prior to</b> Computer Lab Quadrilateral activity in Scratch to explain the use of coordinate grid and ordered pairs in Scratch.  <i>Supports the use of ordered pairs in Scratch.</i>	10-15 minutes  Week of March 14
#3 Math Minilesson		Math Minilesson: Conditionals and Regular/Non-Regular Polygons  (goes with Lesson 11.1 on polygons)	Teach <b>prior to</b> Computer Lab lesson Scratch Quadrilateral Quiz to explain the use of conditional statements with shapes.  <i>Supports math content for Lesson 11.1. Supports use of conditionals in upcoming Computer Lab lesson.</i>	10 minutes  Week of March 21, extends Lesson 11.1
Computer Lab Scratch Card: Scratch Quadrilaterals Quiz	Scratch <a href="#">Quadrilaterals Quiz</a> card		Teach <b>after</b> #4 Math Minilesson on conditionals and quadrilaterals	20-30 minutes  Week of March 21  <b>Exit Ticket</b>
	Create a shapes quiz on polygons using conditionals and My Blocks		<i>Supports the math content in Lesson 11.2 and introduces conditionals and procedures (My Blocks).</i>	
#4 Math Minilesson		Math Minilesson: Conditionals and Quadrilaterals  (goes with Lesson 11.3 on quadrilaterals or after Lesson 11.3)	Teach <b>after</b> Computer Lab Quadrilateral activity in Scratch  <i>Supports math content for Lesson 11.3 Supports use of conditionals in Computer Lab lessons.</i>	10 minutes  Week of March 21, supports Lesson 11.3  <b>Exit Ticket</b>
Computer Lab Scratch Card: Triangle Quiz	Scratch <a href="#">Triangle Quiz</a> card  Create a triangle quiz on equilateral, isosceles, and right triangle types using conditionals		Teach <b>after</b> students learn lesson 11.2 on triangles and after #2 Math Minilesson on conditionals  <i>Supports the math content in Lesson 11.2 and uses conditionals and procedures (My Blocks).</i>	20-30 minutes  Week of March 28  <b>Exit Ticket</b>
		Math Minilesson: Interior/Exterior Angles of	Teach <b>after</b> Computer Lab "Triangle Quiz" to explain the interior/exterior angles in My	10 minutes



SCAN ME



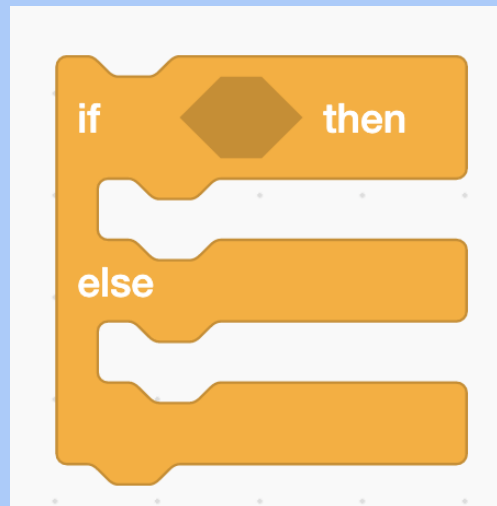
*Image is from Go Math! page 638*

1. What do the polygons in the left circle have in common with each other?
2. What do the polygons in the right circle have in common with each other?
3. What do the polygons in the center section have in common with both groups?

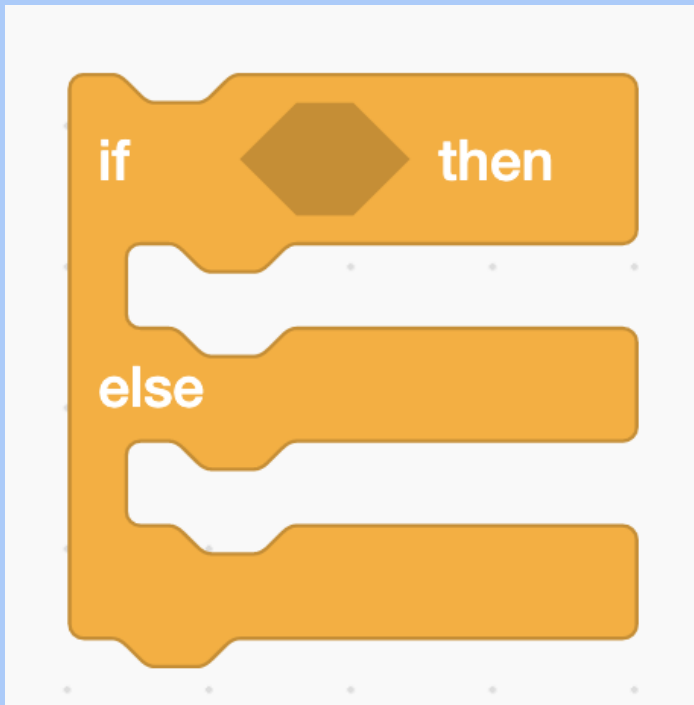
If a triangle is regular, *then* it has three \_\_\_\_\_ sides and angles, *else* it is not regular.

If a quadrilateral is regular, *then* it has four \_\_\_\_\_ angles and sides, *else* it is \_\_\_\_\_.

\_\_\_\_\_ a pentagon has five congruent angles and sides, \_\_\_\_\_ it is a regular pentagon, \_\_\_\_\_ it is not regular.







*If a quadrilateral is regular,  
then it has four \_\_\_\_\_ angles and sides,  
else it is \_\_\_\_\_.*

*Which quadrilaterals are regular quadrilaterals? How do you know?*

Come up with a conditional statement to describe the regular quadrilateral you chose.

# GEOMETRY "SCRATCH CARDS"



[shorturl.at/ayMOR](https://shorturl.at/ayMOR)

Is this a regular polygon or is not a regular polygon? Type 1 for regular polygon or type 2 for not a regular polygon.

Yes, you got it!

No, the correct option was 1: regular polygon because all sides are congruent and all angles are congruent.

The sides of the square are all congruent. This means they are the same length.

### GET READY

You will need these blocks for this step

- Control:
- Operators:
- Sensing:

```
ask "Is this a regular polygon or is not a regular polygon? Type 1 for regular polygon or type 2 for not a regular polygon." and wait
say "Yes, you got it!" for 5 seconds
say "No, the correct option was 1: regular polygon because all sides are congruent and all angles are congruent." for 5 seconds
```

### ADD THIS CODE

```
when green flag clicked
point in direction 90
pen up
set pen size to 5
erase all
go to x: 0 y: 0
pen down
set size to 20 %
square
ask "Is this a regular polygon or is not a regular polygon? Type 1 for regular polygon or type 2 for not a regular polygon." and wait
if answer = 1 then
say "Yes, you got it!" for 5 seconds
else
say "No, the correct option was 1: regular polygon because all sides are congruent and all angles are congruent." for 5 seconds
```

After you connect the Quiz code, nano asks a question. Since a **square** is a **regular polygon**, the correct answer is **1**.

### TRY IT

Click the green flag to start

# EXPANSIVELY FRAMING MULTIPLICATION OF FRACTIONS AND CODE HS

## Unpacking a Fraction Equation: Karel Shows the Product!

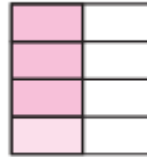


SCAN ME

© Houghton Mifflin Harcourt Publishing Company • Image Credits

**C** The Carter family has only  $\frac{1}{2}$  of a box of cereal at the beginning of the week. They ate  $\frac{3}{4}$  of the  $\frac{1}{2}$  box of cereal.

- Shade the model to show  $\frac{3}{4}$  of  $\frac{1}{2}$  box of cereal.
- Write an expression to show  $\frac{3}{4}$  of  $\frac{1}{2}$  box of cereal.  $\frac{3}{4} \times \frac{1}{2}$
- Will the product be *equal to*, *greater than*, or *less than*  $\frac{1}{2}$ ? *than*  $\frac{3}{4}$ ?

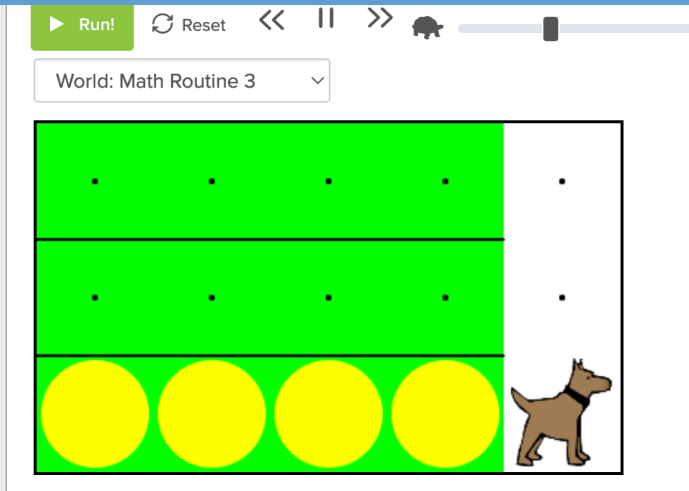


**The product will be less than either factor.**

Chapter 7 445

Why is the Product Getting Smaller?  
 $\frac{1}{3} \times \frac{4}{5} = \frac{4}{15}$

```
Switch to Code View
1 // Starts my code sequence.
2 function start() {
3   // Repeats "putBall, move" sequence 4 times.
4   for (var i = 0; i < 4; i++) {
5     putBall();
6     move();
7   }
8 }
```





# LEARNING EXPONENTS AND SCRATCH CODING

SCAN ME

```
clicked
  set size to 30 %
  erase all
  go to x: -220 y: -160
  set Repeated Addition to 
  set first_number to 4
  set second_number to 3
  set Answer to 0
  set loop count to second_number
  say join Let's look at join first_number join x second_number
  repeat second_number
    set Answer to Answer + first_number
  repeat first_number
```

first\_number 4 second\_number 3

Tera: Answer 12 Ladybug3: Answer 64

Sprite Tera x -220 y -90

Stage

The runtime interface shows a dark stage with a grid of 12 ladybugs (3 rows by 4 columns) and a group of 12 Tera characters (3 rows by 4 columns). The variables 'first\_number' and 'second\_number' are set to 4 and 3 respectively. The 'Tera: Answer' variable is 12, and the 'Ladybug3: Answer' variable is 64. The Tera character is positioned at x: -220 and y: -90.



NSF Grant  
#2031382  
#2031404



### Contact Information:

[kimberly.beck@usu.edu](mailto:kimberly.beck@usu.edu)

[jessica.shumway@usu.edu](mailto:jessica.shumway@usu.edu)

[digitalcommons.usu.edu/eled\\_support](https://digitalcommons.usu.edu/eled_support)

Instructional  
Resources



SCAN ME

These Slides



SCAN ME

## Q&A

### Acknowledgment

This work was supported by National Science Foundation Grant no. 2031382. Opinions, findings, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agency. We thank the participating teachers and students.