LESSON 4

CODING AND COMPUTATIONAL THINKING

KINDERGARTEN

CS FOR ALL

NSF
Award #1837380

SUMMARY AND STANDARDS

Summary:

In this lesson, students will write a story with a beginning, middle, and end, develop an algorithm to create the story, and reenact a story, with a beginning, middle, and end, that you created yourself with an algorithm.

ELA Standards:

Communication and Multimodal Literacies:
1.1 The student will develop oral communication skills.
   j) Ask and respond to questions to seek help, get information, or clarify information.
   l) Increase listening and speaking vocabularies.

Reading:
K.7 The student will expand vocabulary and use of word meanings.
   a)   Discuss meanings of words.
   b)   Increase vocabulary by listening to a variety of texts read aloud.
   c)   Use vocabulary from other content areas.
K.8 The student will demonstrate comprehension of fictional texts.
   b)   Relate previous experiences to what is read
   c)   Use pictures to make predictions.
   d)   Ask and answer questions about what is read.
   e)   Use story elements of characters, settings, and events to retell stories sequentially using beginning, middle, and end.

Writing:
K.11 The student will write in a variety of forms to include narrative and descriptive.
   c) Use letters to phonetically spell words that describe pictures or
      experiences.

CS Standards:

K.1 The student will construct sets of step-by-step instructions (algorithms) either independently or collaboratively including sequencing that emphasize the beginning, middle, and end.

K.2 The student will construct programs to accomplish tasks as a means of creative expression using a block based programming language or unplugged activities, either independently or collaboratively, including sequencing, emphasizing the beginning, middle, and end.

MATERIALS AND RESOURCES NEEDED FOR THIS LESSON:

In preparation for this lesson, you may need a copy of *The Three Little Pigs*

● [Student coding story map](#)

Note: The following lesson is a culminating project in which students write, code, and act out their own adaptation story. You may wish to extend this lesson over two days.

LESSON OBJECTIVES: I CAN...

❏ Ask and answer questions about a story

❏ Write a story with a beginning, middle, and end

❏ Sequence a story with a beginning, middle, and end

❏ Debug an algorithm

❏ Develop an algorithm to represent the beginning, middle, and end of a

story scene

❏ Act out the story algorithm you wrote with a partner

Review the lesson objectives for today's lesson with the students.

COMPUTER SCIENCE

Introduce Computer Science Word Wall and Algorithms: show "computer science", "algorithms", and "motion blocks" word wall cards.

Teacher: "Welcome back to computer science (show "computer science" word wall card) in Kindergarteners!

algorithm

Algorithms are directions for a computer to solve a problem or to do something.

"Today, we will be thinking about **algorithms.** Can you say **algorithm**? (Prompt echo response) **Algorithms** are just like directions. Just like when we play a game and need to follow the directions for how to play, computers need directions, too.

Algorithms are directions for a computer to solve a problem or to do something. People and computers solve problems and tasks every day. Something that makes humans and computers different is that humans can make their own decisions. Computers

need computer programmers to tell them what to do.
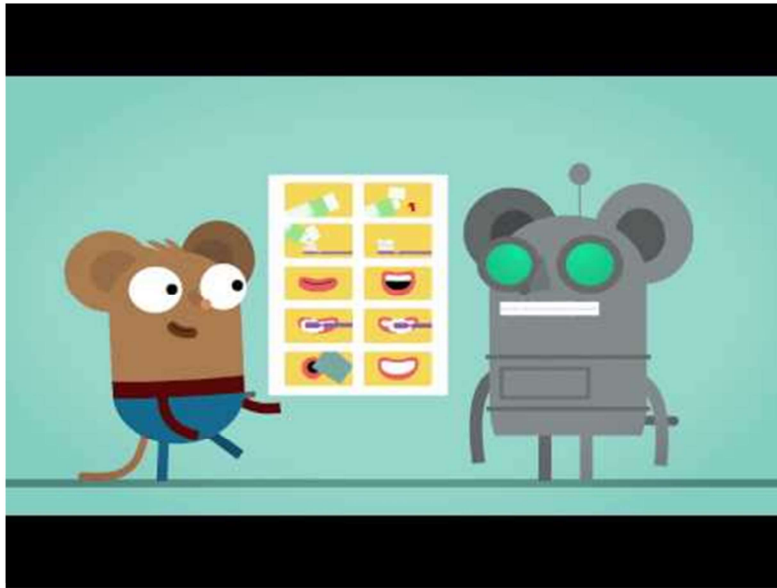
"Now we are going to watch a video. While we are watching, I want you to think about the computer science words that you hear and what they mean. Do you hear words we have talked about when we have been learning about computer science? Do you hear new words?"

What new word did you hear? *Provide students an opportunity to share .* That's right.

Yes, like we mentioned earlier, today is all about **algorithms.** And, the video was ALL about **algorithms**! An **algorithm** is a set of directions!

Video link:

https://www.youtube.com/watch?v=Da5TOXCwLSg

ALGORITHM: SET OF DIRECTIONS; A LIST OF STEPS TO FINISH A TASK

An Algorithm is a list of steps or commands to finish a task.

ALGORITHMS

- Use the correct sequence
- Be clear and precise

FIRST

NEXT

Code: The language that computer scientists create and use to tell a computer what to do.

```
when ⚑ clicked
move 1 steps
move 1 steps
turn ↻ 90 degrees
move 1 steps
move 1 steps
```
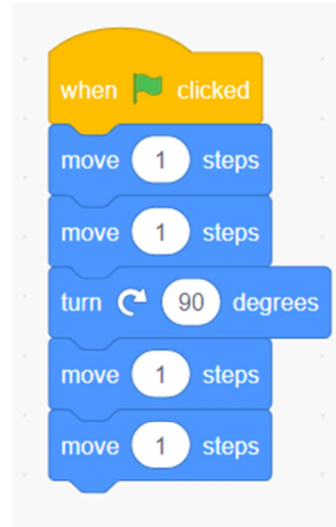
Algorithms must be written in the correct sequence so that others, humans or a computer, can follow the orders to complete a task. This task can be anything, so long as you can give clear instructions for it. Think about an algorithm for getting your shoes on in the morning, you must put on your socks before you put on your shoes!

In computer science, we write our algorithms using code. Code is the instructions that computer scientists create and use to tell a computer what to do.
Writing code is like writing the commands for a computer. When you know how to write code, you can tell computers what to do.

But you can write instructions for humans as well as computers! Recipes are one great example. You can write recipes for:
–how to make hot chocolate
–how to make a yummy treat
–how to make a chocolate, vanilla, or chocolate milkshake!
–how to make koolaid
–how to make lemonade

ALGORITHMS = SET OF INSTRUCTIONS

We may wish to write algorithms for how to....

➢ Get somewhere (directions)
  ○ To the cafeteria
  ○ The park in your neighborhood
➢ Do something (instructions)
  ○ Build a fort in your living room
  ○ Do a dance
  ○ Shoot a basketball or kick a soccer ball
  ○ Create a craft
➢ Explain something
  ○ How your family celebrates the holidays
  ○ About someone important to you or someone famous
  ○ How something happens, such as photosynthesis or the water cycle

There are lots of times when you may wish to explain something to someone! [read slide]

TURN & TALK: HOW WOULD YOU WRITE AN ALGORITHM FOR "GETTING READY FOR SCHOOL"?

Give students time to discuss. Call on a few students for examples.

WE USED BLOCKS TO RETELL OUR STORY!

We can create an algorithm for lots of things, including retelling a story. Here you can see that I created an algorithm to make the itsy bitsy spider go up and down the water spout.

"Look at my algorithm. What did we notice about the spider? Where did Itsy Bitsy go?(*Wait time*). *Provide students an opportunity to share.*

That's right! Itsy Bitsy went up, and then it rained, and the rain washed itsy bitsy down, and then the sun came out and Itsy Bitsy went back up again. Notice that the spider went ALL the way up, so I put the number 10 under my arrow to show that she

moved 10 steps. Then she wen ALL the way down, so I put the number 10 under the down arrow. In the last block the spider only went up half way, so put the number 5 to show that she only moved 5 steps.

ALGORITHM

Algorithm: The green start flag, motion blocks, and end block.

We made an **algorithm** and didn't even realize it!
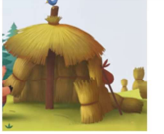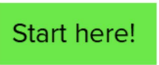
INDEPENDENT PRACTICE

"THE THREE LITTLE PIGS" READ ALOUD
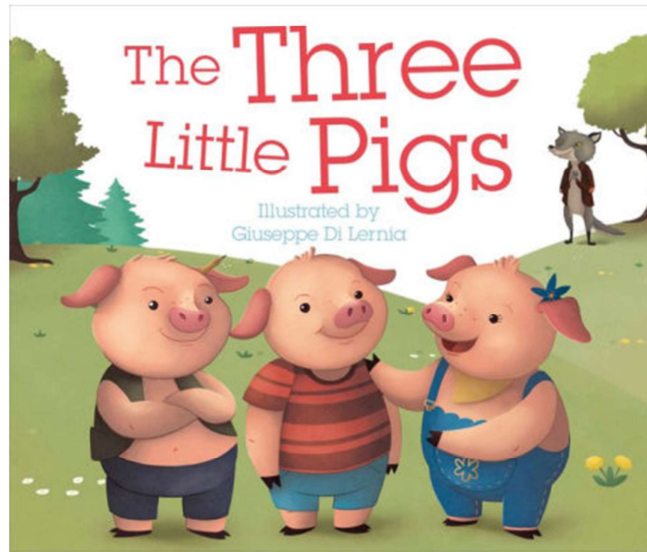
https://www.youtube.com/watch?v=FNYBQsay_Ek

Optional: Play this 5-minute read-aloud of the Three Little Pigs **OR** just remind students of the story.

Each square equals a single step

Today you are going to write an **algorithm** to guide the Big Bad Wolf to visit each of the three little pigs' houses following the **sequence** in the story. You will do this by using code blocks like the blue block on your screen or by drawing arrows on your paper. The instructions are on the next screen.

INSTRUCTIONS: BIG BAD WOLF'S ALGORITHM

Write an **algorithm** to guide the Big Bad Wolf to visit each of the three little pigs' houses following the **sequence** in the story.

You may choose to use either words or coding blocks to write your algorithm.

BONUS: can you find a way to **abstract** any steps in your algorithm to make it shorter and more efficient?

Read slide NOTE: YOU CAN HAVE STUDENTS DO THIS ON PAPER OR USING THE STUDENT SLIDE DECK. EITHER OPTION IS FINE.

WORD & CODE BANKS

Code bank:

Word bank:

- Move left one step
- Move right one step
- Move up one step
- Move down one step
- Turn right one time
- Turn left one time
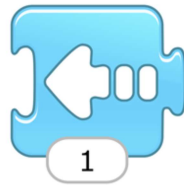
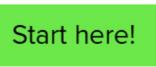These are the blocks you can use to write your algorithm, or you can draw arrows like these on your paper.
NOTE: Have students open student slide deck or complete on paper.

PAIR DEBUGGING!

Swap your neighbor and see if you can spot and debug any errors in their algorithm.

⚠ Each square equals a single step

Here is my solution. Let's act it out. What happens when I follow the arrows? Do I go to every house?

Does yours like like this? Did you choose a different path? Did you remember to turn? What would happen if you followed the code you created?

VOLUNTEERS?

Ask volunteers to act out their algorith and see if they make their way to every house.

WRAP UP

LET'S REVIEW

ALGORITHM

Review algorithms

- Ask students to explain what **algorithms** are to a friend or family member *and* think about another **algorithm** you may want to try.

"Great job, computer scientists! You were working to create algorithms in ScratchJr to make a part of our story come to life. Remember, we are learning to be computer scientists. We can use **algorithms** every day to give computers directions about what to do or to solve problems. Tonight, I want you to explain what **algorithms** are to a friend or family member *and* think about another **algorithm** you may want to try.

YOU CAN BE A COMPUTER SCIENTIST!

Using your **pattern recognition, sequencing, abstraction, decomposition** and **algorithm-writing** skills, you can be a computer scientist!

Remind students that they are computer scientists and writers. They can use computer science skills like finding **patterns, debugging, decomposing, abstracting,** and **coding algorithms** to do important work, like reading and writing stories.

"Today and every day, you are computer scientists and writers. We can use our computer science skills, like finding **patterns, debugging, decomposing, abstracting,** and **coding algorithms** to do important work, like reading and writing stories. You are experts!"

TODAY'S CAREER IN TECH: MEET TESS

Play video: Anyone can be a computer scientist!