FOCUS READERS

Lesson Plan

Thinking Like Machines

Book: *How Coding Works*
Series: Coding
Level: Navigator

Objective

To help students build computational thinking skills by creating sets of step-by-step instructions to guide a partner through a maze.

Supplies

- *How Coding Works* book
- Whiteboard
- Maze handout (attached)
- Paper and pencils

Before the Activity

Have students read *How Coding Works* before class. Print a copy of the Maze handout for each pair of students. Cut it in half so Maze A and Maze B are on separate sheets of paper.

Activity

Choose a student to read Chapter 1 ("What Is Code?") out loud. Ask students to define the word *code* (Answer: A set of instructions that tells a machine what to do). Explain that a robot's computer can only do exactly what it is told—nothing more.

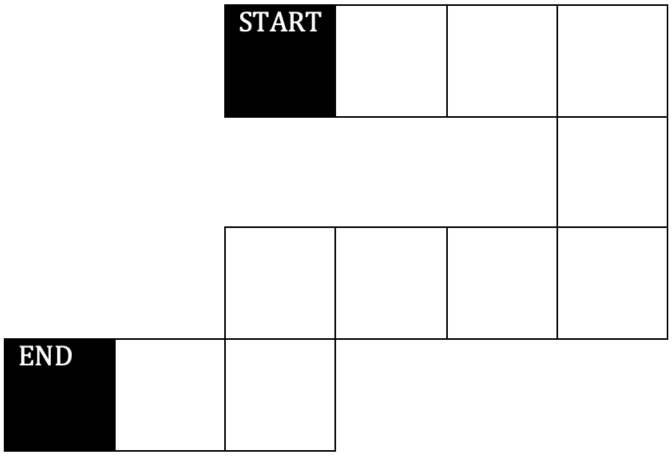
In order to do an action, a robot needs a list of instructions. This list of instructions is sometimes called an algorithm. If the algorithm doesn't include a step, the robot will not know to do it.

To write instructions for robots, people must practice thinking like machines. They must anticipate the kinds of instructions a robot will need, and they must write those instructions in a way the robot can understand.

To practice, ask students to imagine they are trying to guide a robot through a maze. The robot can understand 6 commands:

- move 1 square forward
- move 1 square backward
- turn 90° to the right
- turn 90° to the left
- pick up a ball
- set down a ball

Draw the following maze diagram on the whiteboard:



Ask, "If a robot is standing on the square labeled START and facing the right, what is the first command we should give the robot?" If students say anything other than "move 1 square forward," demonstrate what the robot would do and allow students to correct their mistake. Then ask students what command would come next. Continue until students have guided the robot to the square marked END.

Explain that the students just created an algorithm for the robot—a list of instructions telling the robot how to complete the task of going through the maze. Changing the algorithm into symbols the robot understands is known as programming. For instance, you could use the following symbols to guide the robot through the maze:

- move 1 square forward = →
- move 1 square backward = ←
- turn 90° to the right = ↪
- turn 90° to the left = ↩
- pick up a ball = ↑
- set down a ball = ↑

Using these symbols, the instructions students gave you would look like this:
→ → → ↪ → → ↪ → → → ↩ → ↪ → →

Have students find pairs to practice writing programs with these symbols. Give one student in each pair Maze A, and give the other student Maze B. On a separate piece of paper, each student should draw arrows to create a program to that will guide a robot through their maze. The program should include steps that tell the robot to pick up the ball and put it in the basket. When students finish writing instructions, students should trade papers and take turns testing each other's instructions. One student will be the robot, following the instructions exactly as the other student watches and looks for errors. Then the students should trade roles. Students should fix any errors in their program and test it again.

When both students in each pair have a successful program, come back together as a class. Explain that the arrows are a shorter way to write the program than using words, but programmers have another way to make programs even shorter: loops. A loop tells a computer to repeat a step or set of steps. For instance, have students examine the instructions for the maze you drew on the whiteboard, looking for repeated steps:
→ → → ↪ → → ↪ → → → ↩ → ↪ → →

Using loops, this program could be written this way:
$\overset{3}{\enclose{circle}{\rightarrow}} \hookrightarrow \overset{2}{\enclose{circle}{\rightarrow}} \hookrightarrow \overset{3}{\enclose{circle}{\rightarrow}} \hookleftarrow \rightarrow \hookrightarrow \overset{2}{\enclose{circle}{\rightarrow}}$

Now, instead of writing 15 arrows, the programmer only needs to write 9 arrows. Have students look for repeating patterns in their programs. Have students rewrite each program using circles and numbers to represent loops. Once again, students in each pair should trade and test each other's programs until both programs are successful.

Evaluation
Use the attached answer key to check students' answers (and to provide prompting and support to students who struggle with writing instructions). During the testing phase, remind students to follow their partner's instructions exactly as written, even if the instructions don't work.

Standards
This lesson may be used to address the Common Core State Standards' speaking and listening standards, grade 5 (SL 5.1), and the National Science Education Standards' Content Standard E, grades 5–8.

Answer Key

Maze A

Answer: → → → → → → ↪ → → ↩ → → ↑ ↪ → → → ↪ → → → → ↓ ← ↩ → → ↩ → → → → (32 arrows)

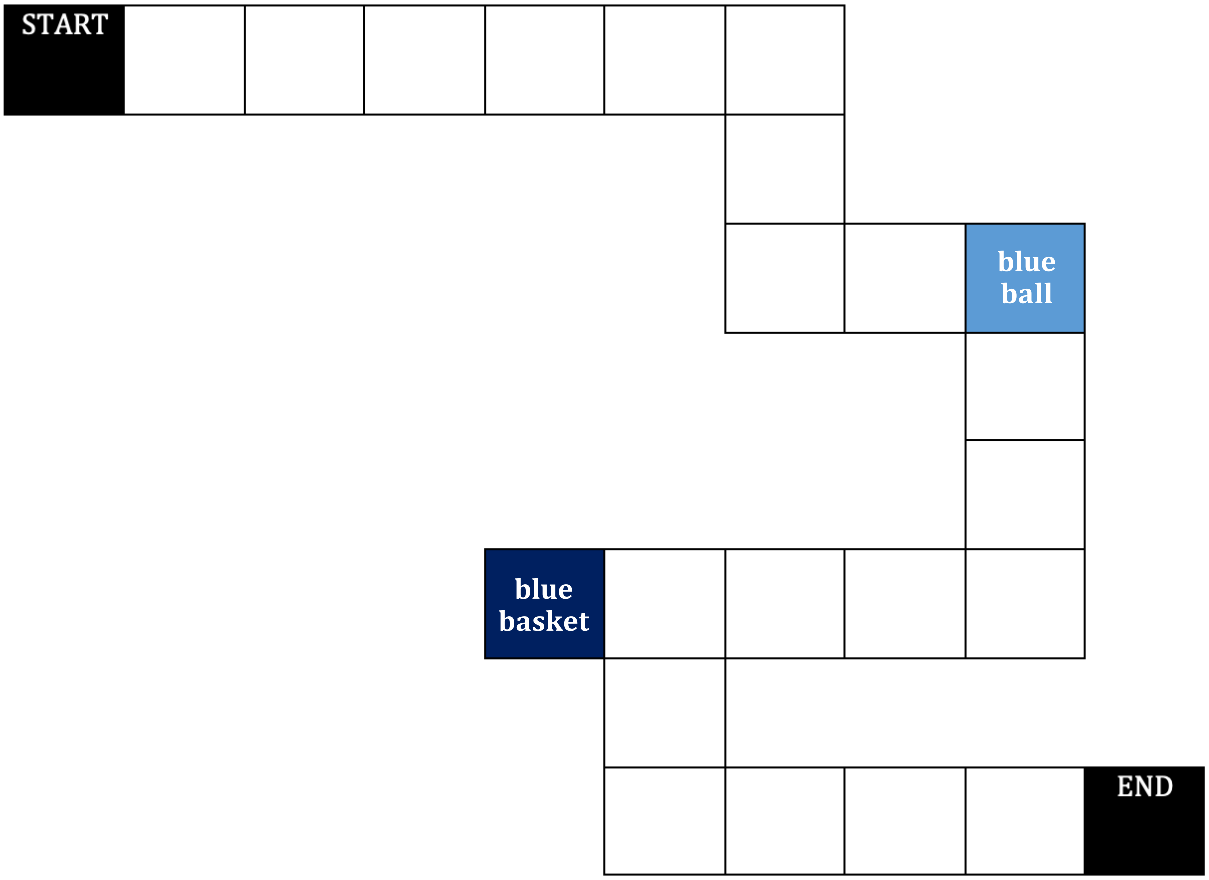
Answer with loops: $\overset{6}{\textcircled{\rightarrow}}$ ↪ $\overset{2}{\textcircled{\rightarrow}}$ ↩ $\overset{2}{\textcircled{\rightarrow}}$ ↑↪ $\overset{3}{\textcircled{\rightarrow}}$ ↪ $\overset{4}{\textcircled{\rightarrow}}$ ↓ ← ↩ $\overset{2}{\textcircled{\rightarrow}}$ ↩ $\overset{4}{\textcircled{\rightarrow}}$ (16 arrows)

Maze B

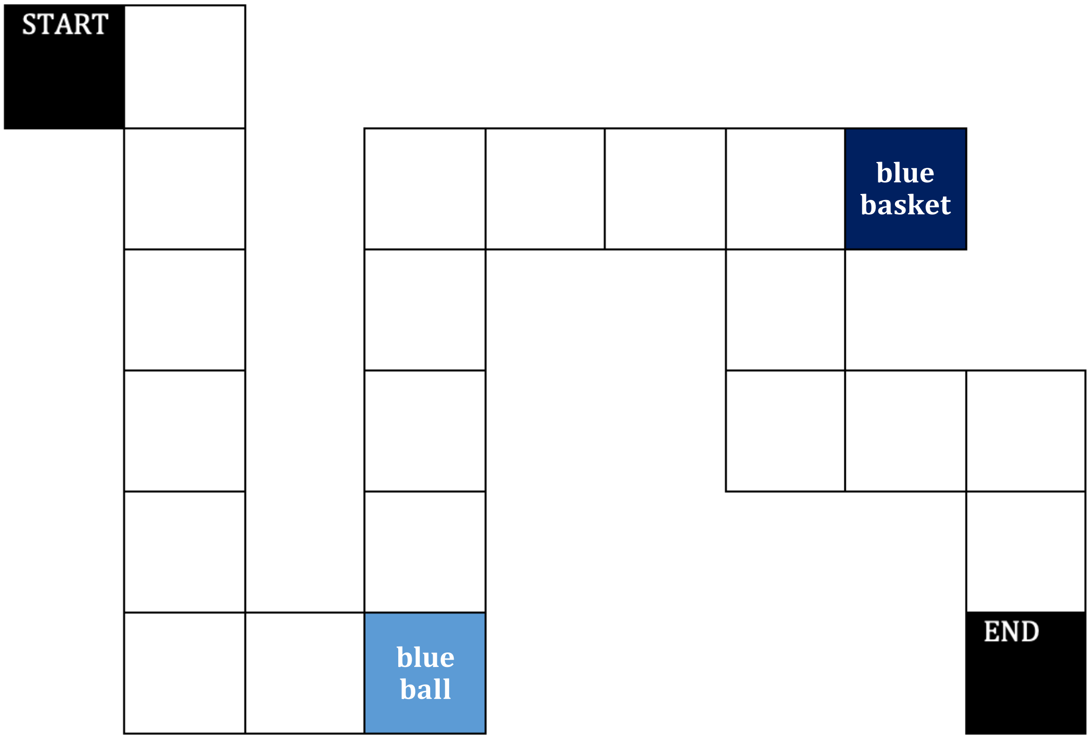
Answer: → ↪ → → → → → ↩ → → ↑ ↩ → → → → ↪ → → → → ↓ ← ↪ → → ↩ → → ↪ → → (32 arrows)

Answer with loops: → ↪ $\overset{5}{\textcircled{\rightarrow}}$ ↩ $\overset{2}{\textcircled{\rightarrow}}$ ↑ ↩ $\overset{4}{\textcircled{\rightarrow}}$ ↪ $\overset{4}{\textcircled{\rightarrow}}$ ↓ ← ↪ $\overset{2}{\textcircled{\rightarrow}}$ ↩ $\overset{2}{\textcircled{\rightarrow}}$ ↪ $\overset{2}{\textcircled{\rightarrow}}$ (18 arrows)

Maze A


START
blue ball
blue basket
END

Maze B


START
blue basket
blue ball
END