PROGRAM SIMULATOR

READY

GRADES: 7-12 DURATION: 30-60 min. EXPERIENCE: Beginner SUBJECT: Computer Science

MATERIALS

- Handouts 1-3
- Pencil (optional)
- Black marker
- Bit or Evo (optional)

VOCABULARY

PROGRAMMING LANGUAGE: Specific vocabulary, symbols, and rules a computer understands. There are several "languages" computers understand, such as JavaScript, Python, HTML, and more.

PROGRAM: A list of instructions written in a programming language a computer understands and executes.

STATEMENT: A single action carried out sequentially as part of a program. This can be compared to a sentence in a paragraph.

PSEUDOCODE: Code intended to be read by a human rather than a computer. A simplified version of a programming language.

SET

Today, you will be using pseudocode to simulate what a computer reads as it executes commands. You get to be the program! Let's review the rules of this pseudocode that was made just for this challenge:

Algorithm Type	Statement End Indicator	Code Block Indicators	Keywords
Sequential *Commands are executed in the same order they are written in your program. Hint: computers are very literal when reading a program. Don't actually move until you read a command specifically telling you to!*	Period (.) *Periods seperate different commands.* *Note: Many existing programming languages use a semi-colon (;) to indicate the end of a statement.*	Brackets ({...}) *Brackets let you know that there will be several commands that need to be executed together before moving on.*	If...do...else *If you see these words, commands are to be executed based upon a specific condition.* Repeat... *This keyword indicates that a command needs to be performed multiple times.*

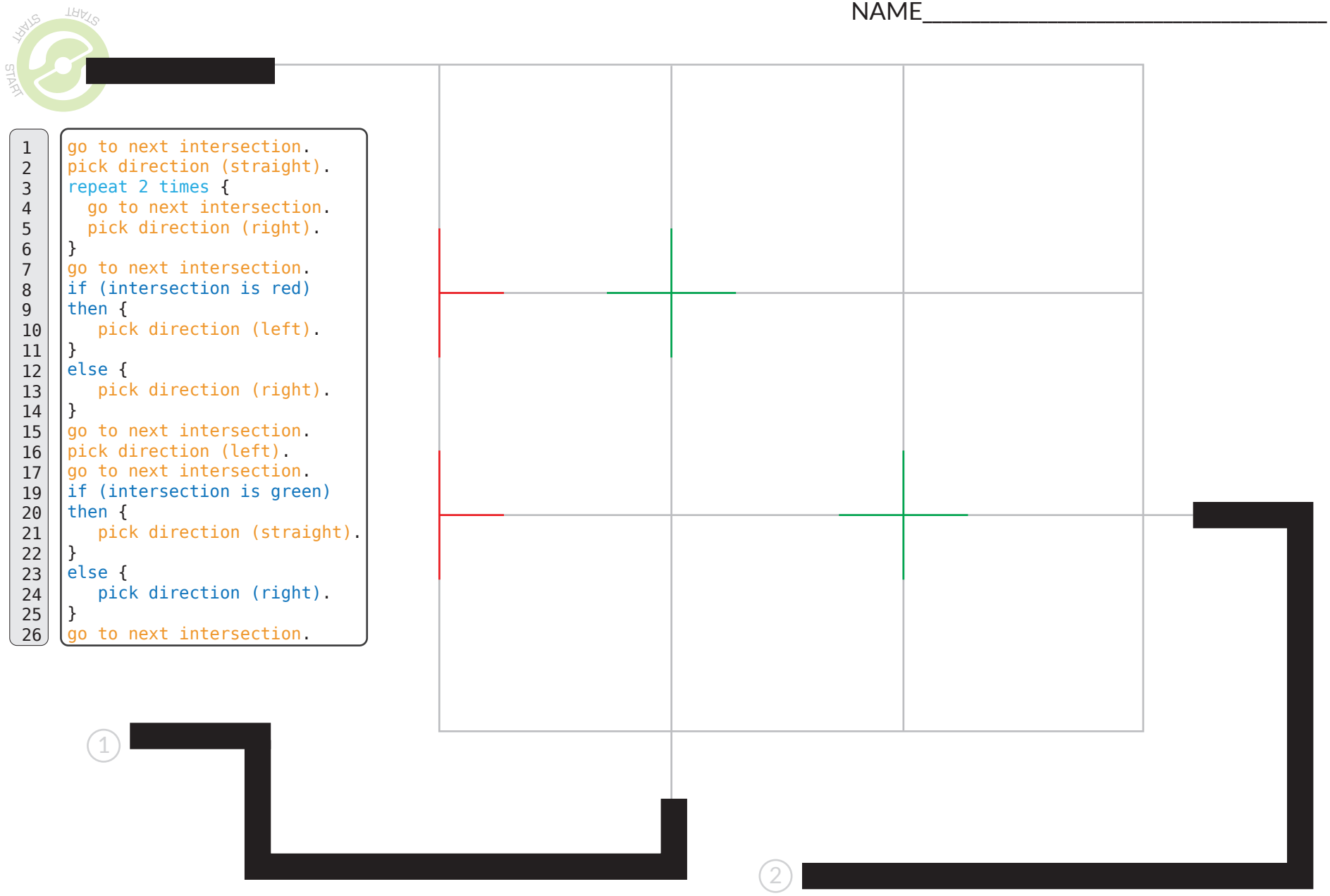
Optional: Allow students to try Program Simulator 1 before explaining these principles. After they are done, discuss what they think indentation, brackets, periods, etc. mean and then try Program Simulator 2, which is more difficult.

CODE!

The student will be the computer, now! They will complete Program Simulators 1 & 2 (1 is easy/intermediate, 2 is advanced), reading the program and executing commands. The final activity requires the students to write a program based on what they have learned! Solutions are provided on the final pages.

NAME_____

```
1  go to next intersection.
2  pick direction (straight).
3  repeat 2 times {
4    go to next intersection.
5    pick direction (right).
6  }
7  go to next intersection.
8  if (intersection is red)
9  then {
10    pick direction (left).
11 }
12 else {
13    pick direction (right).
14 }
15 go to next intersection.
16 pick direction (left).
17 go to next intersection.
19 if (intersection is green)
20 then {
21    pick direction (straight).
22 }
23 else {
24    pick direction (right).
25 }
26 go to next intersection.
```

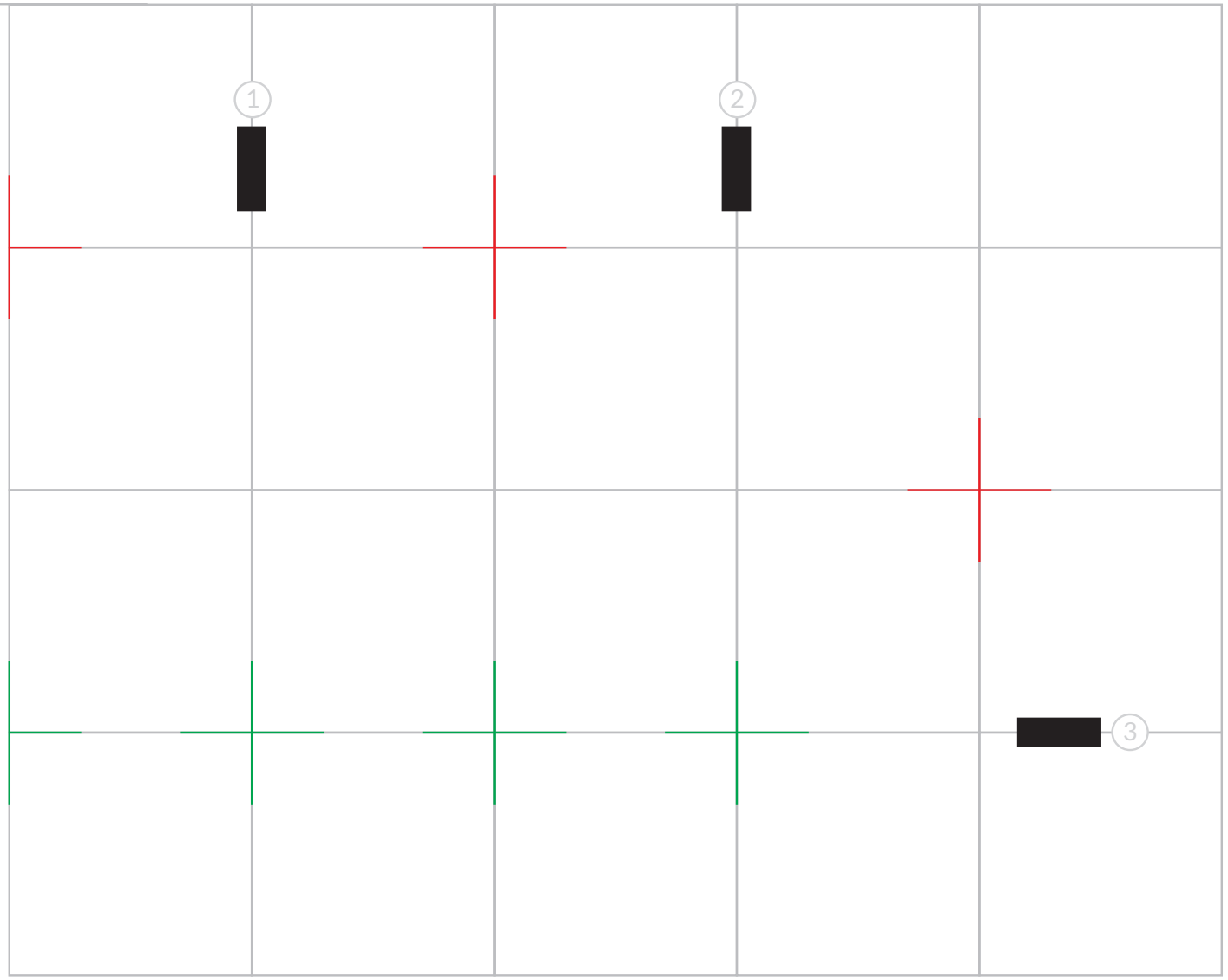


ozobot PROGRAM SIMULATOR 1

Do: Simulate a computer program by executing the commands in the box. Use a marker and see if you end up at the correct number. You may want to use a pencil first.

NAME_____

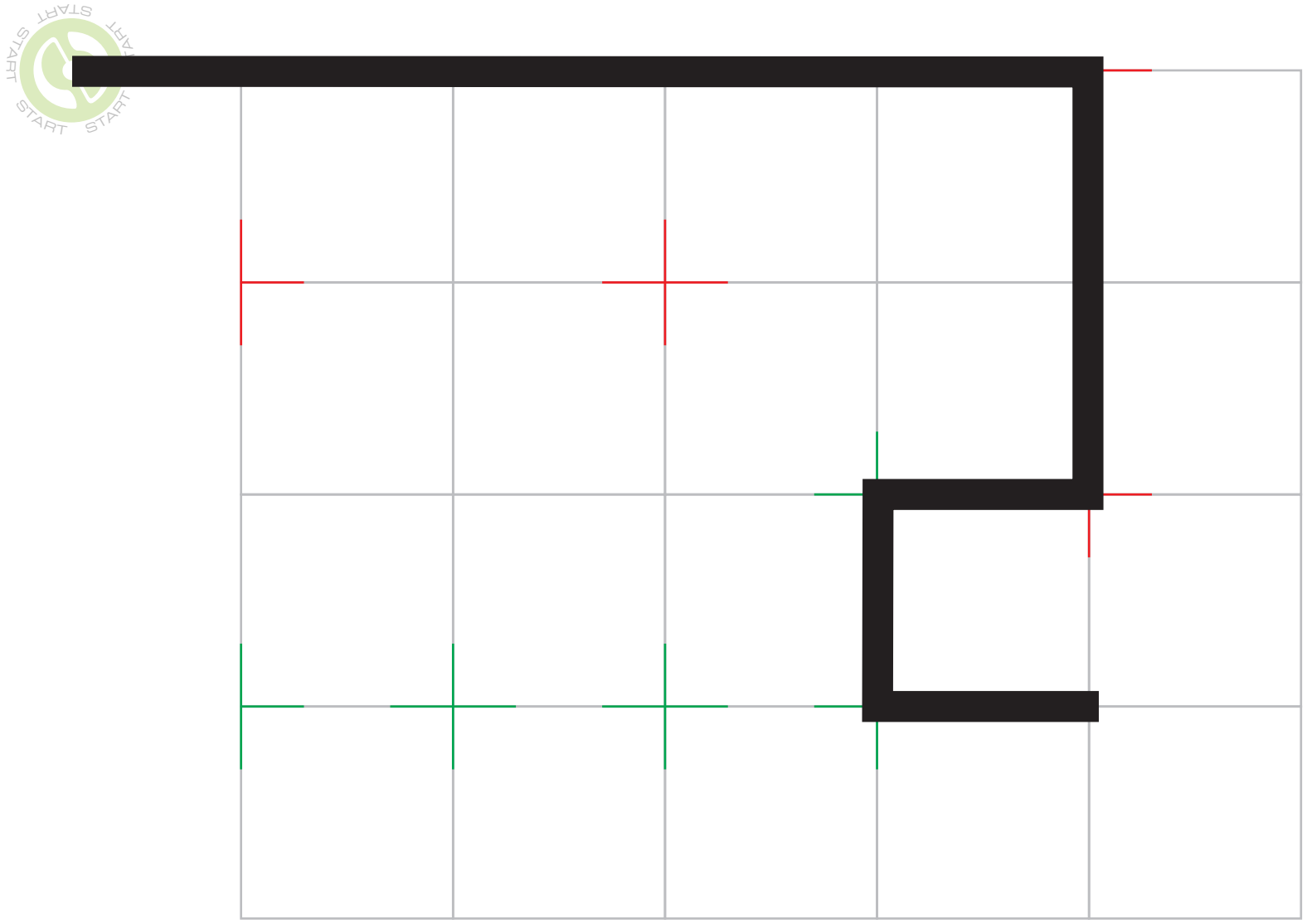

```
1  repeat 2 times {
2     pick direction (straight).
3     go to next intersection.
4  }
5  pick direction (right).
6  go to next intersection.
7  if (intersection is red)
8     then {
9  go to next intersection.
10 }
11 else {
12    pick direction (left).
13 }
14 go to next intersection.
15 pick direction (left).
16 go to next intersection
17 }
19 go to next intersection.
20 repeat forever {
21 if (intersection is green)
22    pick direction (left).
23    go to next intersection
24 }
25 else {
26    stop repeating forever.
27    }
28 }
29 repeat 2 times {
30    pick direction (straight).
31    go to next intersection.
32 }
```



ozobot PROGRAM SIMULATOR 2

Do: Simulate a computer program by executing the commands in the box. Use a marker and see if you end up at the correct number. You may want to use a pencil first.

NAME _____

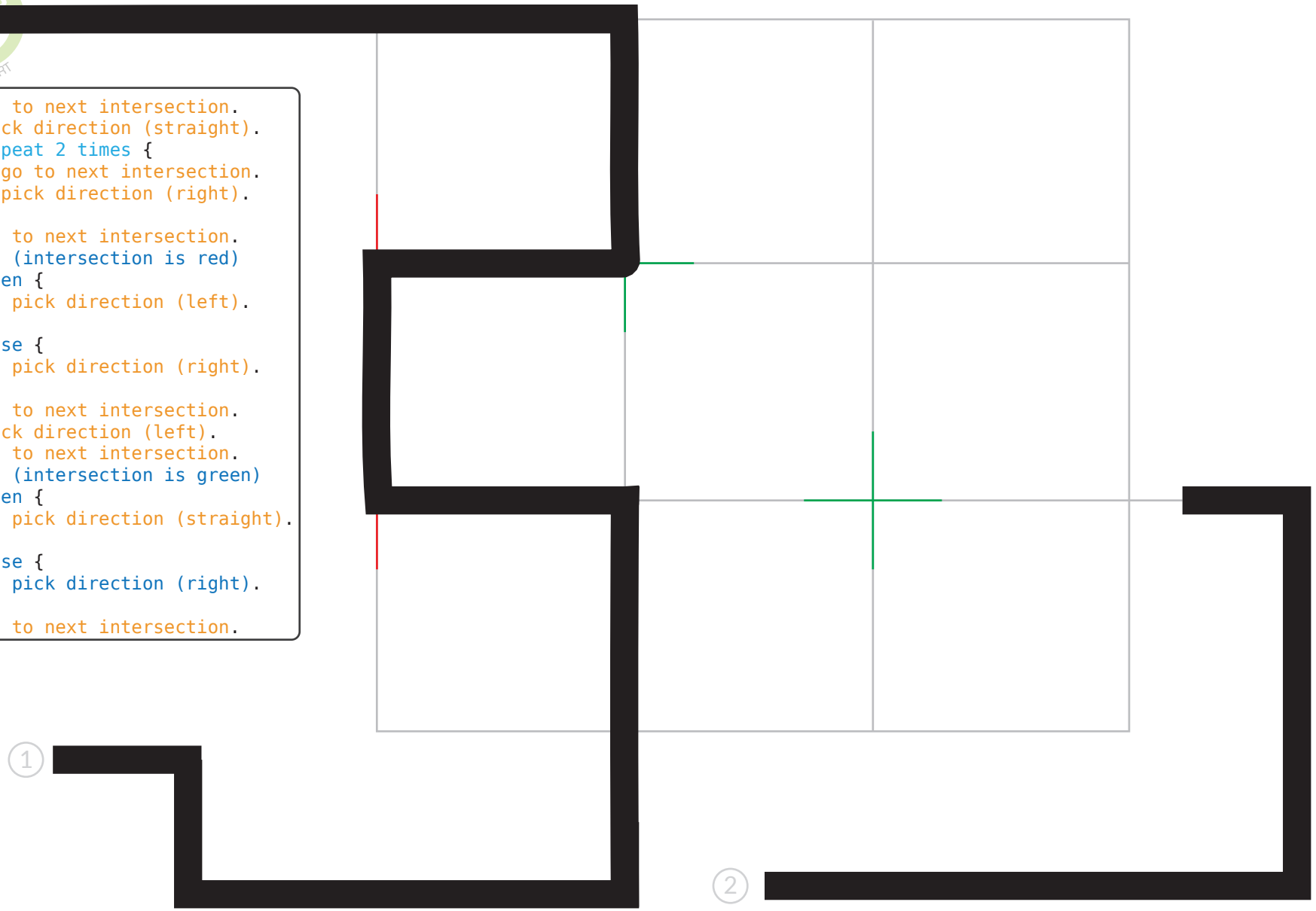


ozobot PROGRAM SIMULATOR

Do: Now that you have practiced reading code and simulating a program, try writing pseudocode for the program above! Note: there are several solutions. It is possible to complete this exercise using only 12 lines of code. Try your best to use the fewest lines possible!

SOLUTION

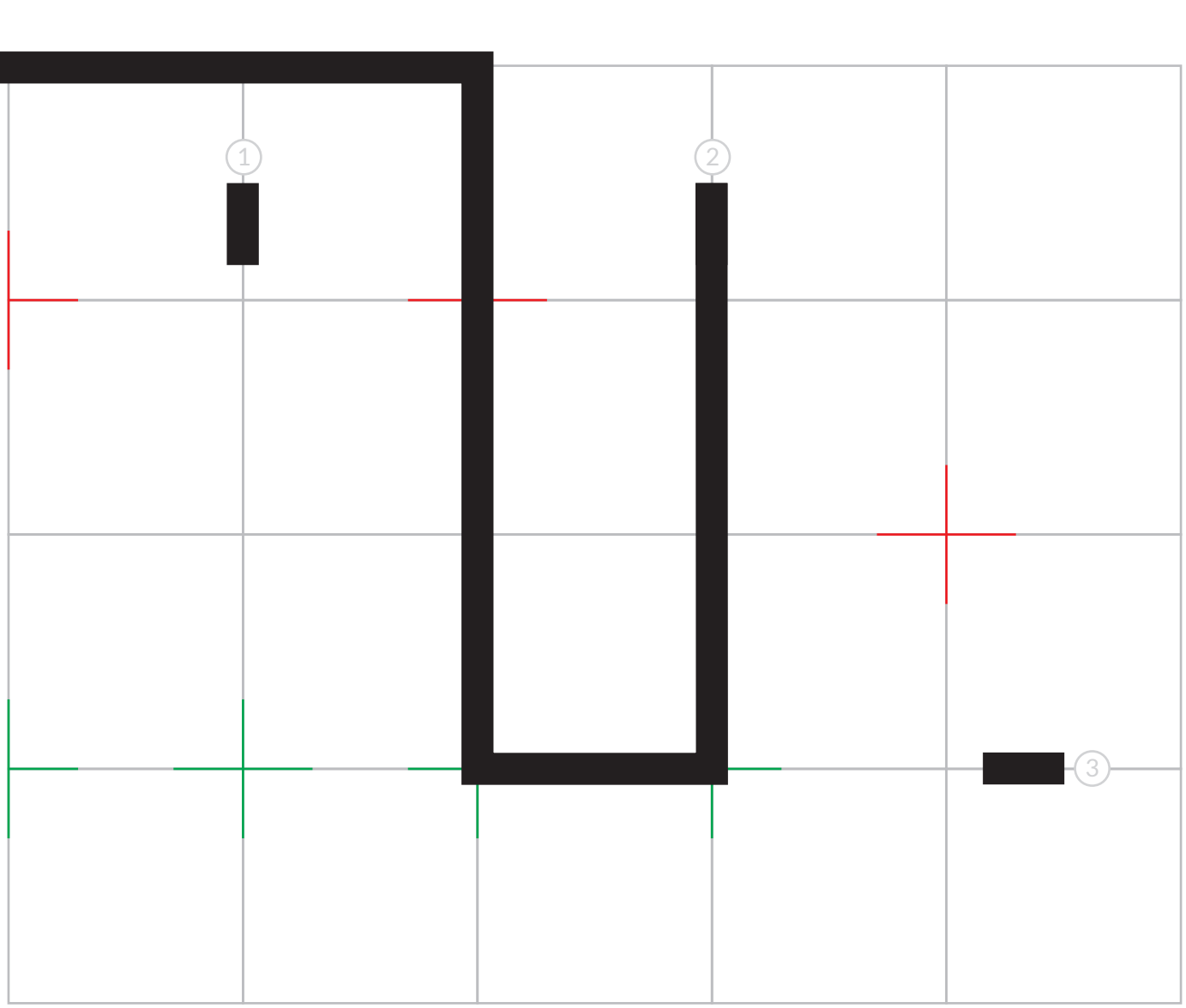
```
1  go to next intersection.
2  pick direction (straight).
3  repeat 2 times {
4    go to next intersection.
5    pick direction (right).
6  }
7  go to next intersection.
8  if (intersection is red)
9  then {
10    pick direction (left).
11 }
12 else {
13    pick direction (right).
14 }
15 go to next intersection.
16 pick direction (left).
17 go to next intersection.
19 if (intersection is green)
20 then {
21    pick direction (straight).
22 }
23 else {
24    pick direction (right).
25 }
26 go to next intersection.
```



ozobot PROGRAM SIMULATOR 1

Do: Simulate a computer program by executing the commands in the box. Use a marker and see if you end up at the correct number. You may want to use a pencil first.

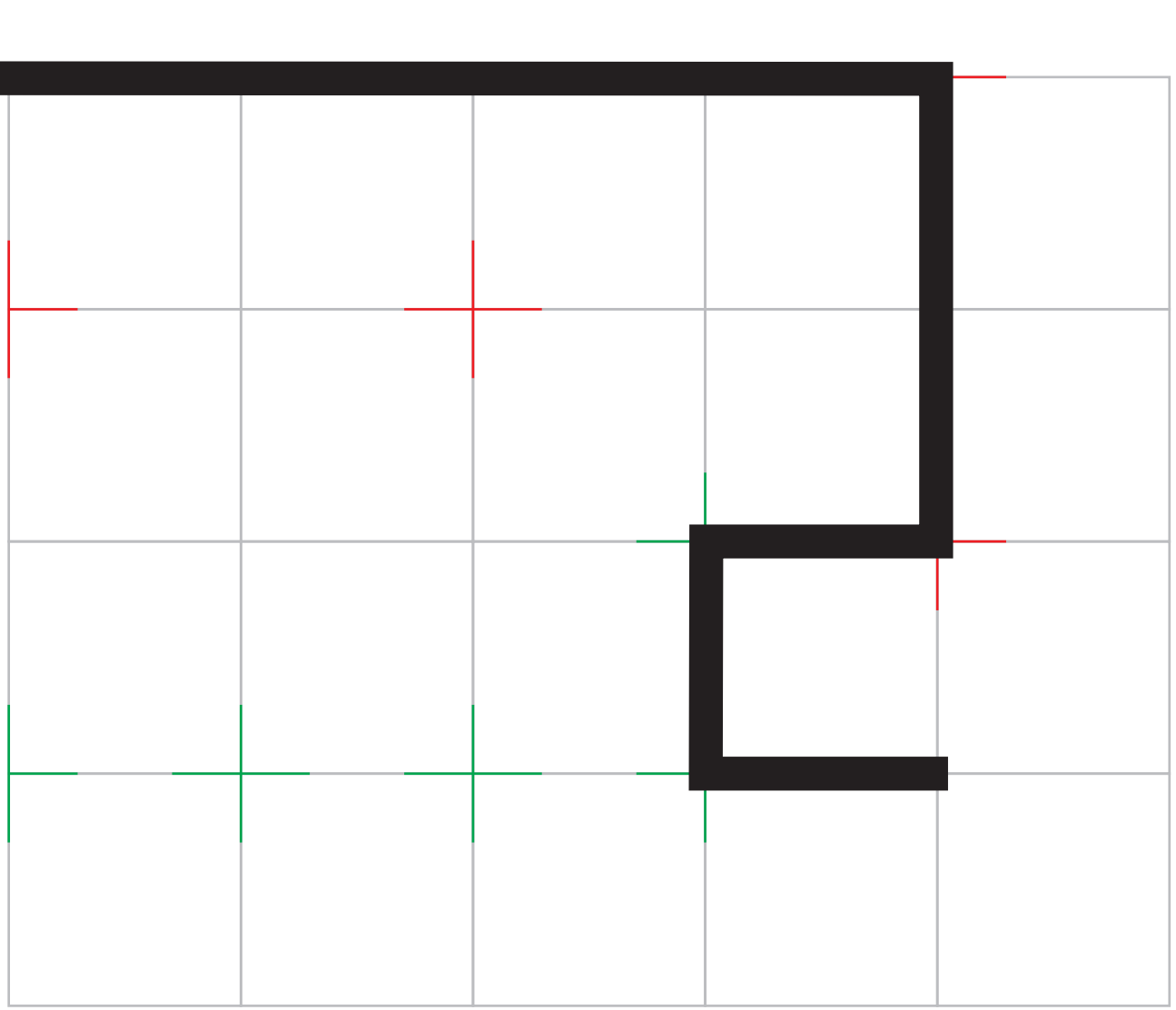
SOLUTION




```
1   go to next intersection.
2   repeat 2 times {
3      pick direction (straight).
4      go to next intersection.
5   }
6   pick direction (right).
7   go to next intersection.
8   if (intersection is red)
9      then {
10  pick direction (straight).
11  }
12  else {
13     pick direction (left).
14  }
15  go to next intersection.
16  pick direction (straight).
17  go to next intersection
18  repeat forever {
19  if (intersection is green)
20        pick direction (left).
21        go to next intersection
22     }
23     else {
24        stop repeating forever.
25        }
26  }
27  repeat 2 times {
28     pick direction (straight).
29     go to next intersection.
30  }
```


Do: Simulate a computer program by executing the commands in the box. Use a marker and see if you end up at the correct number. You may want to use a pencil first.

SOLUTION



```
1   repeat 10 times {
2           go to next intersection.
3           if (intersection is red) {
4                   pick direction (right).
5           }
6           else if (intersection is green) {
7                   pick direction (left).
8           }
9           else {
10                  pick direction (straight).
11          }
12  }
```

ozobot PROGRAM SIMULATOR

Do: Now that you have practiced reading code and simulating a program, try writing pseudocode for the program above! Note: there are several solutions. It is possible to complete this exercise using only 12 lines of code. Try your best to use the fewest lines possible!