

# CREATE AN OZOBOT GAME LEVEL

AN OZOBOT TUTORIAL FOR THE HOUR OF CODE™

PREPARED FOR OZOBOT  
BY LINDA MCCLURE, M. ED.

## Essential Question

What is computer programming?

How can we program a game level for Ozobot?

## Hour of Code™

What is the Hour of Code? The Hour of Code is a nationwide initiative by Computer Science Education Week ([www.csedweek.org](http://www.csedweek.org)) and Code.org ([www.code.org](http://www.code.org)) to introduce millions of students to one hour of computer science and computer programming. The idea is to introduce the concept of coding to students and letting them know that anyone can write computer code!

The following is a planner created by Ozobot that suggests a lesson that can be used for the Hour of Code. In this lesson you will introduce the concept of programming and allow students to practice coding by creating their own game level modeled after the OzoBlockly Games, a fun tutorial game for Ozobot.

No prior programming experience is required for this lesson. It is self-contained and introduces students to programming through OzoBlockly, which is a visual programming language and editor based on Google's Blockly. Please find more information about the Hour of Code here <https://hourofcode.com>.

## Information

The OzoBlockly games (<http://games.ozoblockly.com>) will teach the OzoBlockly language and general programming concepts. Both students who have had experience with Blockly or Scratch and those who have not will enjoy all levels of the tutorials. The practice will prepare students to create their own programs, load the program onto Ozobot and run the program.

The simulated version of Ozobot that is available in the OzoBlockly games is very good at following commands exactly. The real Ozobot Bit is very good at following commands too. However, the real Ozobot Bit may not turn as precisely as the simulated Ozobot. This is related to the wheels. It is important for students to understand that the real Ozobot Bit will approximate the movements, especially the turns. Students' program for the robot may need to compensate for this. Can they build something into their program to help the Ozobot Bit reach the goal?

### **Prerequisites**

No programming experience or knowledge of Ozobot is required.

### **Standards**

- ISTE
- 1.c – Use models and simulation to explore complex systems and issues
  - 4.b – Plan and manage activities to develop a solution or complete a project
  - 6.a – Understand and use technology systems
- CC Math
- 1. Make sense of problems and persevere in solving them.
  - 2. Reason abstractly and quantitatively.

### **Grouping**

Groups of two or three students

### **Materials**

- Tablet or computer with the Ozobot Games website, <http://games.ozoblockly.com> and the OzoBlockly editor <http://ozoblockly.com/>
- Ozobot Bit or Evo, 1 per group of 2-3 students
- Blank paper for sketching

### **Age/Grade Level**

Grade 2 and up. Older students may move more independently through the levels of the tutorials. The tutorials are appropriate for all levels to learn the Ozoblockly programming language.

### **OzoBlockly Programming Topics**

Free Movement, Light Effects

### **OzoBlockly Mode**

Use mode 2 or higher

## Lesson Duration

1 hour

## Vocabulary

- *Ozobot Bit/Evo* - Little robot that can follow drawn lines or can be programmed using visual codes or through the OzoBlockly programming language
- *OzoBlockly* - A visual editor which allows to create programs by plugging blocks together. The blocks can be used to control Ozobot's behavior like movement, LED lights, etc.
- *Rotate Right or Left* - Approximate 90° turn
- *Rotate Slightly Right or Left* - Approximate 45° turn
- *Steps* - a unit of movement forward
- *Line Following* - Ozobot's default capability of sensing and following lines on paper or digital screens
- *Free Movement* - Ozobot's capability of moving around freely without visual guidance such as following lines

## Overview

1. Introduction. Students are being introduced to programming by doing a brainstorming exercise with the entire class.
2. Guided Activity. Students play the tutorial games 1-10 to introduce them to OzoBlockly and the following programming concepts.
  - Visual programming basics
  - Movement forward
  - Turn or rotate
  - Setting LED colors
  - Loading Ozobot and running the program
3. Independent Activity. Students will write their own tutorial level in the Ozoblockly Editor.

## Note

*Levels 4, 7, and 10 are Ozobot only levels. Students must use the Ozobot to test and run the completed program. The simulation feature is disabled.*

## Tips

For more info on how to plan your event for the Hour of Code, please take a look at the Hour of Code website <https://hourofcode.com/how-to/events>.

You can also sign up your event there. Other great pages to help you plan are:

<https://hourofcode.com/how-to>

<https://hourofcode.com/promote/resources>

## **Beyond this Lesson**

After completing this lesson, students should be comfortable writing their own OzoBlockly program. As the next step, you may want to do another lesson using OzoBlockly. You can see all of these lessons in the OzoBlockly section of the Ozobot Lesson Library here <http://portal.ozobot.com/lessons/list/ozoblockly>.

## **Contact**

Please contact us at [ozoedu@ozobot.com](mailto:ozoedu@ozobot.com) with any question about Ozobot or this lesson plan.

## ACTIVITY

### 1. Introduction to Coding

*"How many of you have heard of computer code? Today we are going to learn what coding is, and actually write our own computer code!"*

1. As a class, brainstorm the steps for walking in a square
  - a. Start by allowing the kids to come up with ideas without input from you.
  - b. Write the steps on white board, cards or paper strips so they can be moved around and edited instructions inserted.
  - c. As the steps develop, question the students so they see where the steps need to be more specific. Add details to the instructions.
2. Show video: <https://www.youtube.com/watch?v=THOEq5soVpY> about "What is coding?", or any of the inspirational videos from the Hour of Code website <https://hourofcode.com/us/promote/resources#videos>.

### 2. Guided Class Activity

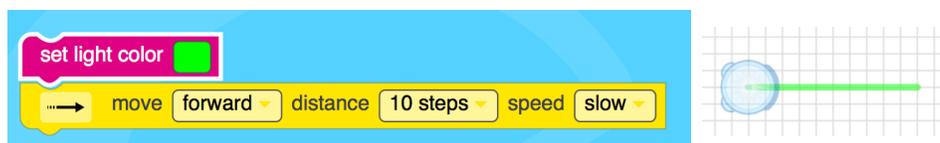
*"Today we are going to write a computer program to teach a robot to do a task"*

Have students go to <http://games.ozoblockly.com>. Note that, for your convenience, we have included the solutions at the end of this lesson plan. Guide your students through the first two levels. After that they should be able to complete the rest of the levels on their own.



#### Level 1

Show the class level 1. Demonstrate to the class how Ozobot's image can move forward or back using a block from "Movement". The units of measurements are "steps". Show the students how to set and change the LED light color using the block from "Light Effects".



**Tip:** When running the simulator, the program is also being checked and feedback is given about the correctness and optimality of the program. Sometimes a solution is correct, but it may be programmed with less blocks. Students can go back and modify their program to make it more optimal. Be aware that sometimes blocks can hide out of sight. These “invisible” blocks can make your program incorrect or inefficient. To avoid this, make sure to click on the trashcan in the bottom right corner before starting a new program. This will delete all blocks in the editor – even the hidden ones.



## Level 2

Explain that Ozobot can make turns. For this we use the “Rotate” block found in “Movement”.

Have the students experiment with movement and rotation. Question the students, “how far does the Ozobot move when programmed to turn a “slight left” versus a “left” turn?”

Discuss 45° rotation versus 90° rotation. Challenge the students to find how many degrees are in a U-Turn.

Students solve level 2

After students have discussed and learned the basic movement and programming, **allow students to complete the 10 levels.**

Note that an Ozobot Bit or Evo will be necessary to run the program on levels 4, 7, and 10 as the simulator is disabled. This gives the students practice calibrating and loading the program onto Ozobot. Click the “Help” button for more information on how to load Ozobot.

**Note:** A game will be unlocked when students solve all 10 levels. This game can be played with a group of students and helps them practice the concepts they have learned so far. Playing the game may be an interesting and fun activity to do at this point if you have the time.

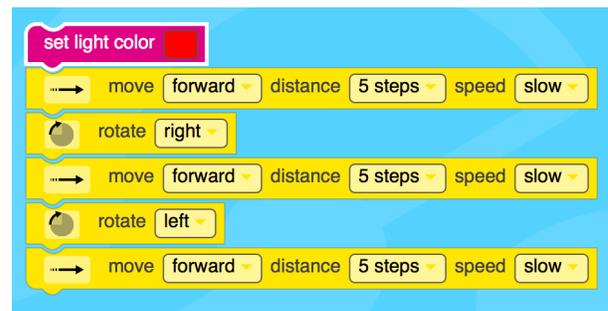


Figure 1: Solution to Level 2

### 3. Independent Activity: Create Level 11

#### Objective

Students will write their own game level 11.

#### Procedure

Now that the students have had a little practice with Ozoblockly, it is their turn to write a game.

1. Have the students sketch a path on paper to be the basis of Level 11. The path should be slightly more difficult than levels 1-10.

Suggested Rules:

- The path should change colors at least twice
  - Ozobot should have to double back on at least one path to complete the level
  - The turns must be 45°, 90°, 180° or a combination of these
2. Go to <http://ozoblockly.com/editor> on a computer or tablet.
  3. Choose "Beginner" (2) or "Intermediate" (3) levels
  4. Have students write the program.
  5. Students test program with Ozobot.

*Note: Students should not put Ozobot on the lines that they drew. Ozobot should be able to complete the task without line following i.e. with blocks from the "Movement" category and not from the "Line Navigation" category. The path the students drew is merely an abstract sketch.*

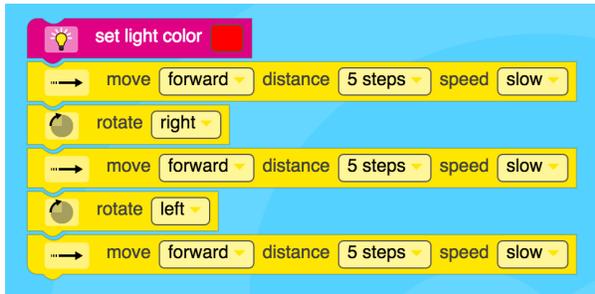
### 4. Tutorial Games Solutions

#### Shape Tracer 1

Level 1

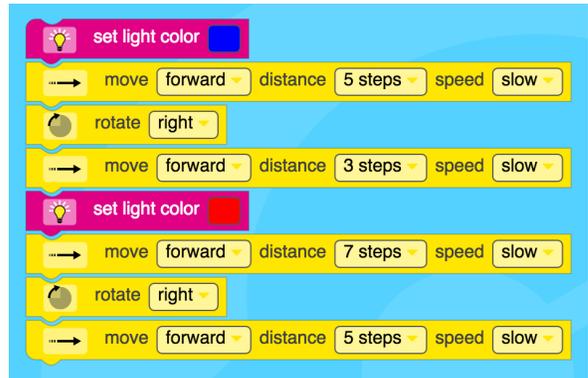


### Level 2



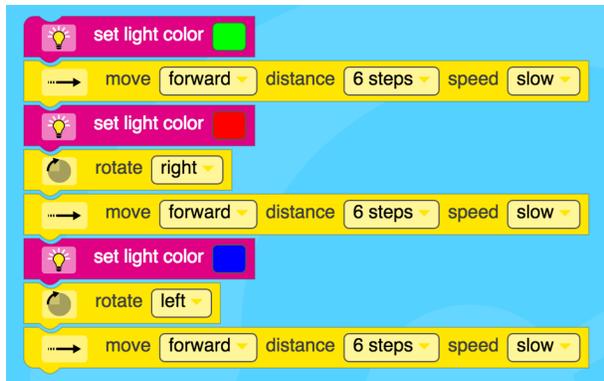
```
set light color [red]
move forward distance 5 steps speed slow
rotate right
move forward distance 5 steps speed slow
rotate left
move forward distance 5 steps speed slow
```

### Level 3



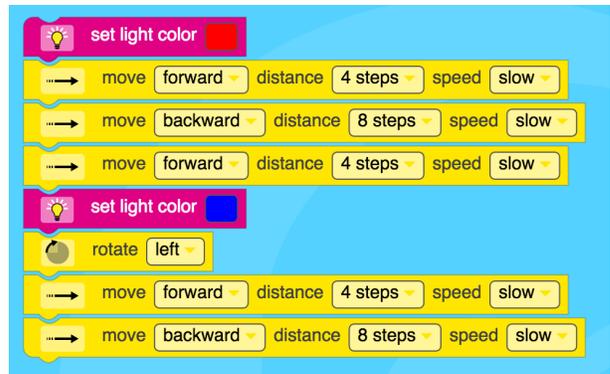
```
set light color [blue]
move forward distance 5 steps speed slow
rotate right
move forward distance 3 steps speed slow
set light color [red]
move forward distance 7 steps speed slow
rotate right
move forward distance 5 steps speed slow
```

### Level 4



```
set light color [green]
move forward distance 6 steps speed slow
set light color [red]
rotate right
move forward distance 6 steps speed slow
set light color [blue]
rotate left
move forward distance 6 steps speed slow
```

### Level 5



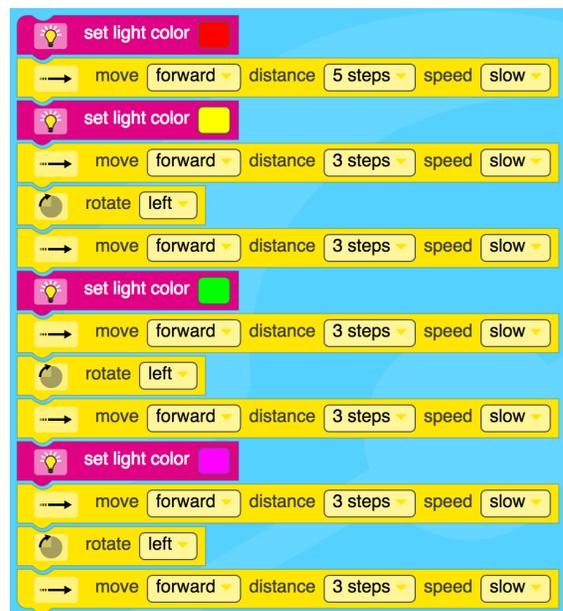
```
set light color [red]
move forward distance 4 steps speed slow
move backward distance 8 steps speed slow
move forward distance 4 steps speed slow
set light color [blue]
rotate left
move forward distance 4 steps speed slow
move backward distance 8 steps speed slow
```

### Level 6



```
set light color [green]
move forward distance 4 steps speed slow
rotate left
move forward distance 4 steps speed slow
rotate left
move forward distance 8 steps speed slow
rotate left
move forward distance 4 steps speed slow
rotate left
move forward distance 4 steps speed slow
set light color [red]
rotate left
move forward distance 4 steps speed slow
```

### Level 7



```
set light color [red]
move forward distance 5 steps speed slow
set light color [yellow]
move forward distance 3 steps speed slow
rotate left
move forward distance 3 steps speed slow
set light color [green]
move forward distance 3 steps speed slow
rotate left
move forward distance 3 steps speed slow
set light color [magenta]
move forward distance 3 steps speed slow
rotate left
move forward distance 3 steps speed slow
```

### Level 8

```

set light color [red]
rotate [slight right]
move forward distance 5 steps speed slow
rotate [right]
move forward distance 5 steps speed slow
set light color [blue]
move forward distance 5 steps speed slow
rotate [left]
move forward distance 5 steps speed slow
rotate [left]
move forward distance 5 steps speed slow
rotate [left]
move forward distance 5 steps speed slow
set light color [red]
move forward distance 5 steps speed slow
rotate [right]
move forward distance 5 steps speed slow
    
```

### Level 9

```

rotate [left]
set light color [magenta]
move forward distance 8 steps speed slow
rotate [slight right]
set light color [yellow]
move forward distance 6 steps speed slow
rotate [slight right]
set light color [magenta]
rotate [right]
move forward distance 8 steps speed slow
set light color [yellow]
rotate [slight right]
move forward distance 6 steps speed slow
    
```

### Level 10

```

rotate [slight right]
set light color [green]
move forward distance 5 steps speed slow
rotate [slight right]
move forward distance 5 steps speed slow
rotate [slight right]
set light color [red]
move forward distance 5 steps speed slow
rotate [right]
rotate [slight right]
move forward distance 5 steps speed slow
rotate [slight right]
move forward distance 5 steps speed slow
    
```