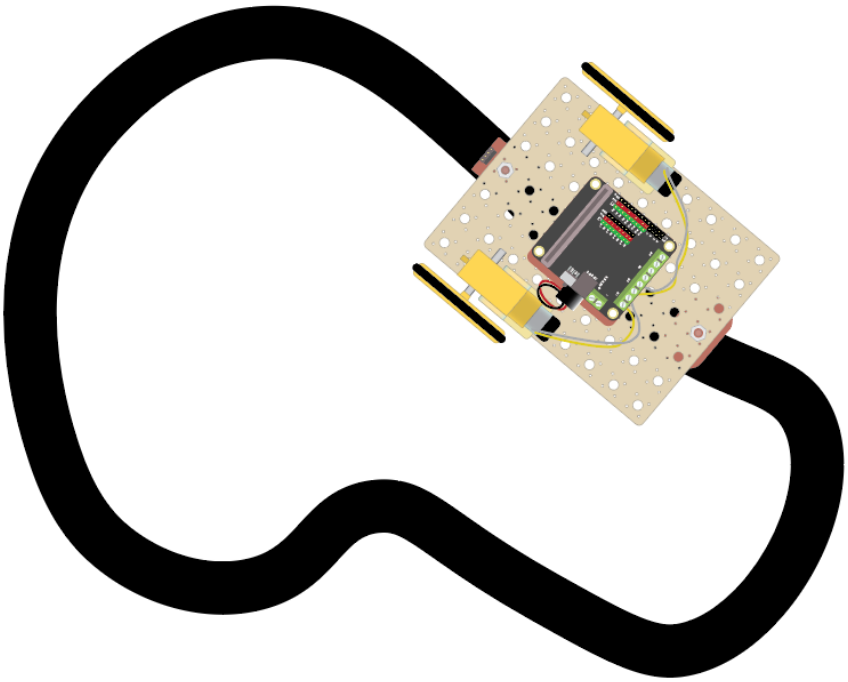
Build a Line Following Robot

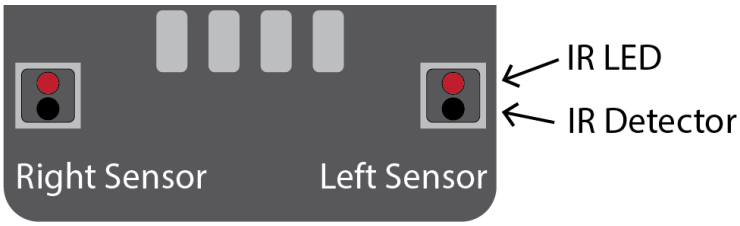
Project 1.06

In this workshop you will make robot that can follow a black line on the floor.

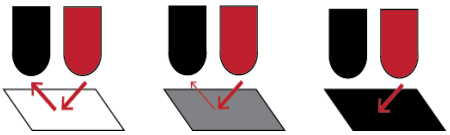


How it Works

You will attach a **line following sensor** to the underside of your robot. The line following sensor is actually made up of two light sensors and two lights:



The light is an **Infrared (IR) light**. This is a light that can't be seen by humans, but can be detected by an **IR sensor**. The light will reflect off a white surface, but with darker surfaces, less light is reflected. A totally black surface will not reflect any light. The IR detector will detect this difference in reflected light →.



By having two sensors you can get your robot to find where the line is. For example, if the left sensor is seeing a dark surface and the right sensor is seeing a light surface then the robot can deduce that the line is to the left of the robot and take the appropriate action.

What to do

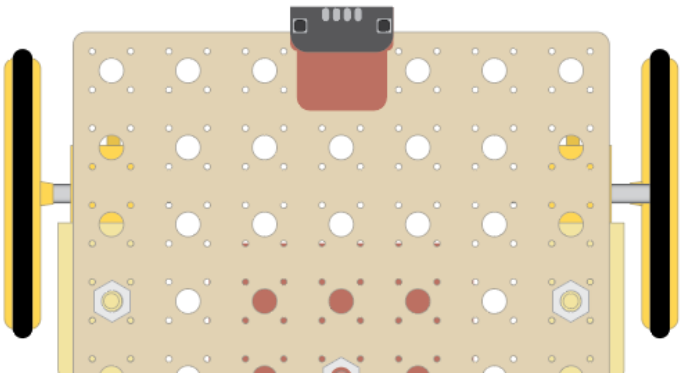
- If you haven't already done so, build the robot by referring to the previous worksheet (just build it, don't code it).
- Then follow this worksheet to add a line following sensor and code it to detect the line
- Finally, attempt the coding challenges to get your robot to follow a line on the line track

Add the Line Following Sensor

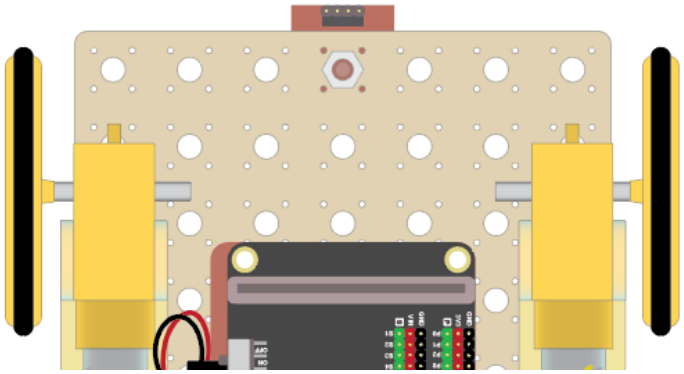
Connect the Line Following Sensor

Connect the line following sensors to the underside of your robot. It's important to connect it in the front and middle of the robot.

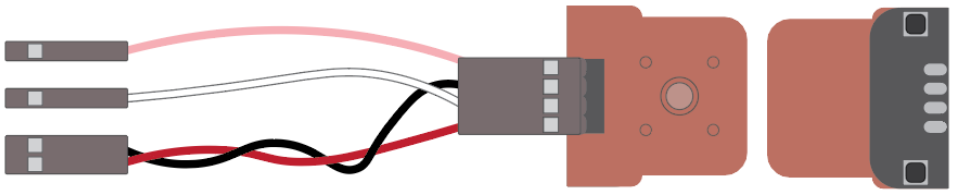
1 Under the robot: Connect the sensor



2 On top: Secure with a nut.

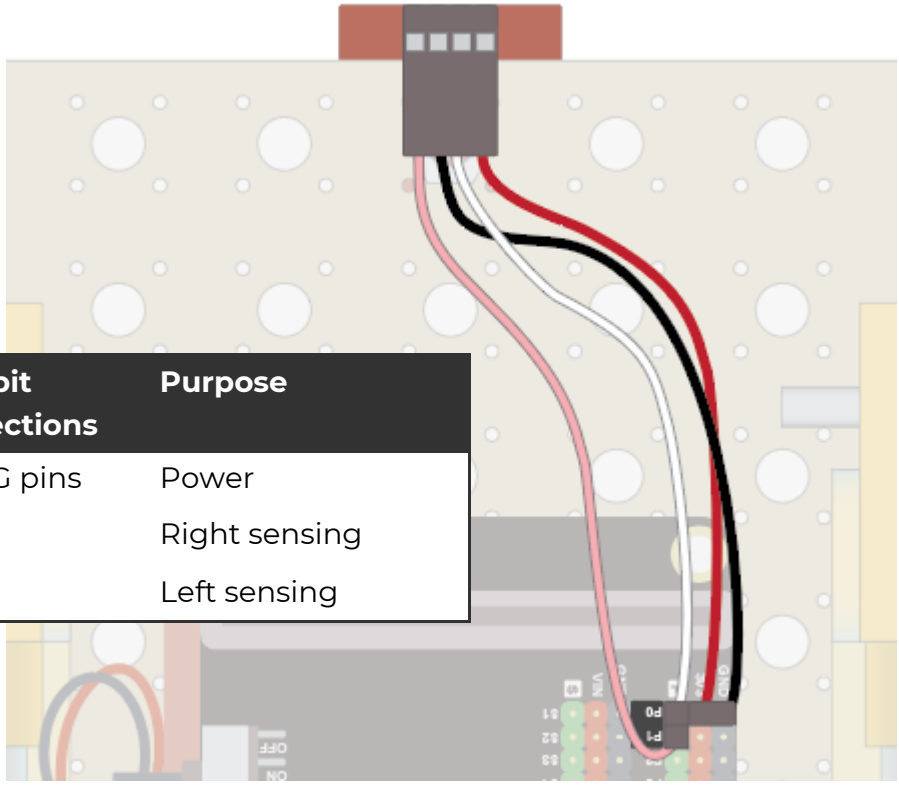


3 Connect the special line follower cable to the sensor



top view bo

4 Wire up the line follower as follows



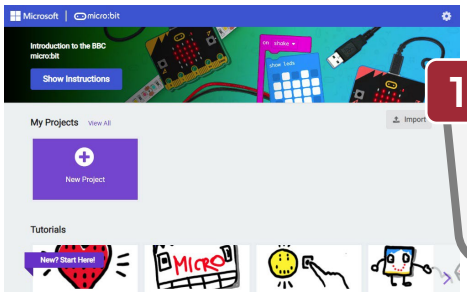
Line Follower	Microbit Connections	Purpose
Red/black	Any VG pins	Power
Right (white)	P0	Right sensing
Left (pink)	P1	Left sensing

Code the Line Following Sensor 1

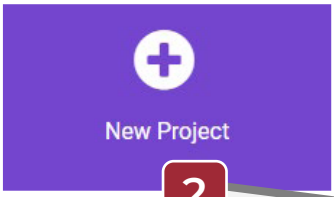
Create a Project

Create a new Makecode project so you can start coding.

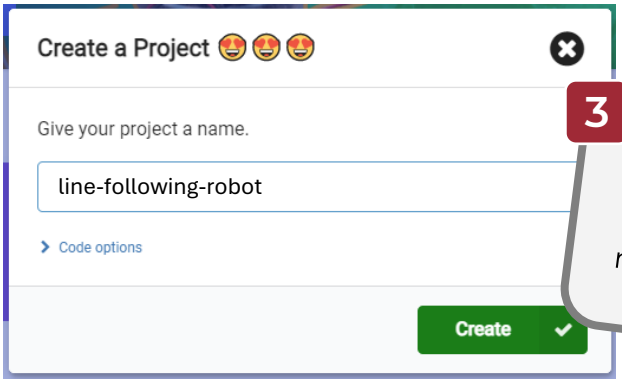
https://makecode.microbit.org/



1 Go to the Makecode website.



2 Click on New Project.

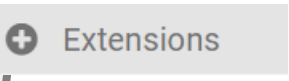


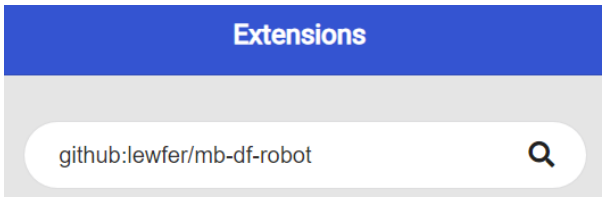
3 Give the project a name (whatever name you want!).

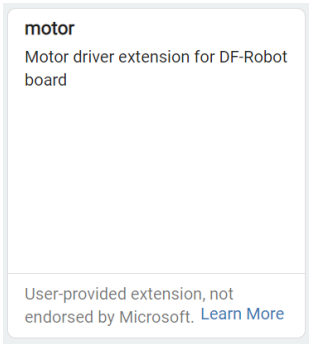
Add the Motor Driver Extension

The motor driver extension gives you the ability to control motors.

3 Add the motor driver extension







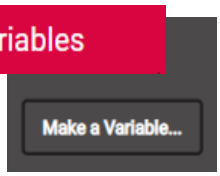
Code the Line Following Sensor 2

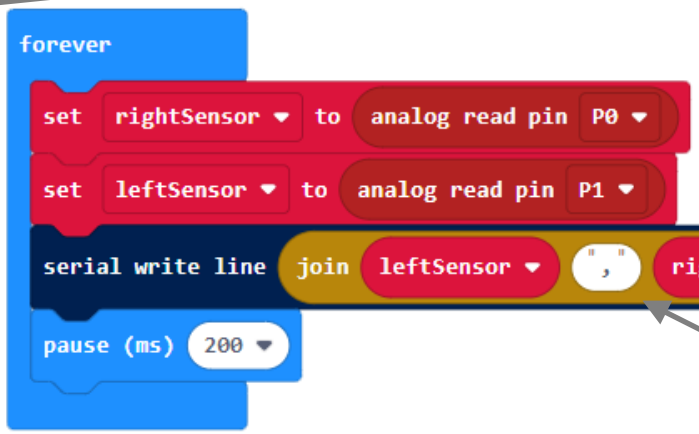
Read the Sensor Values

First we will take readings from the sensors to see how much light they are detecting.

1 Add this code.

You will first need to create two variables: **rightSensor** and **leftSensor**

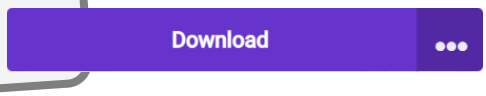




These blocks read the light level from the two sensors.

This block will send the two readings back to your computer.

2 Download the code to the Microbit.

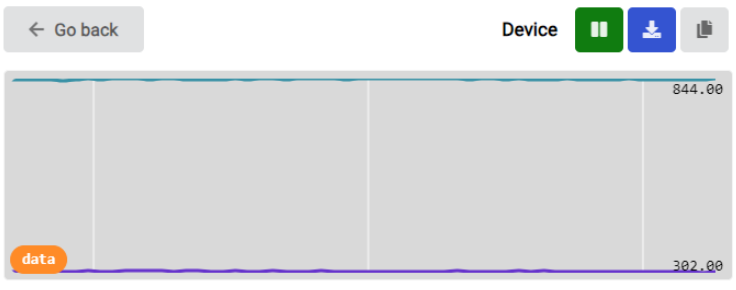


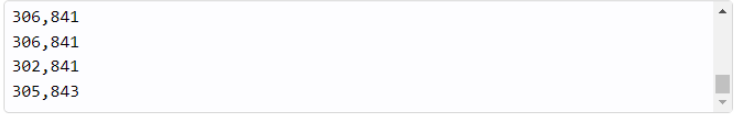
3 To see the data, click on the **Show data Device** button on the left.

Ensure your Microbit is still connected and paired to your computer or you won't see the data.

4 You should see some numbers and a graph appear. These show the left and right sensor readings.

High values mean the sensor sees a light colour, low values mean the sensor sees dark colours.



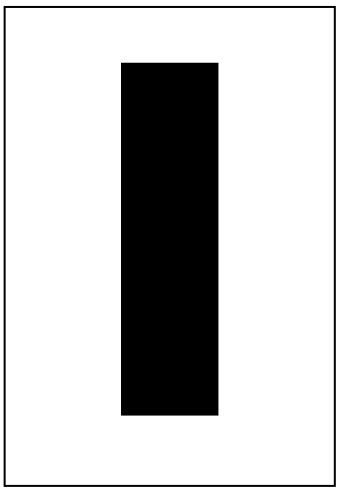


Code the Line Following Sensor 3

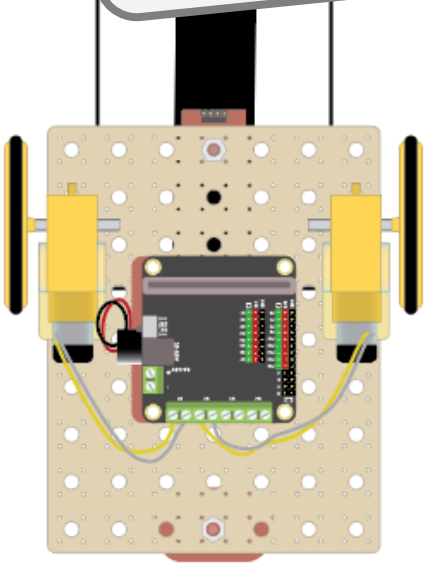
Understand the Readings

Now you have the readings from the left and right light sensors. Before we add some more code, let's understand what these readings mean and think through how we can use them to get the robot to follow a line.

1 Get some paper and draw a thick line on it. The line should be about 6cm long and 3cm wide

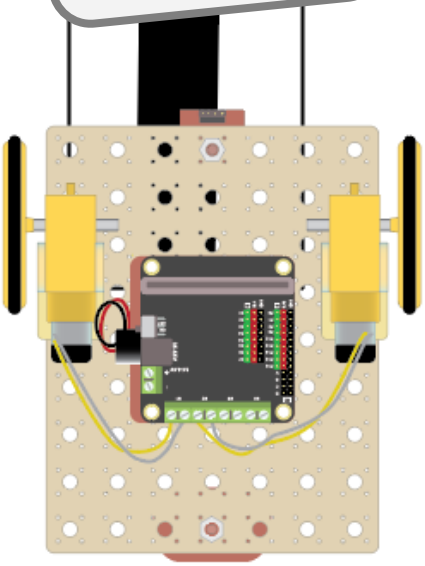


2 Now place the robot on the line, so that both sensors are on the line



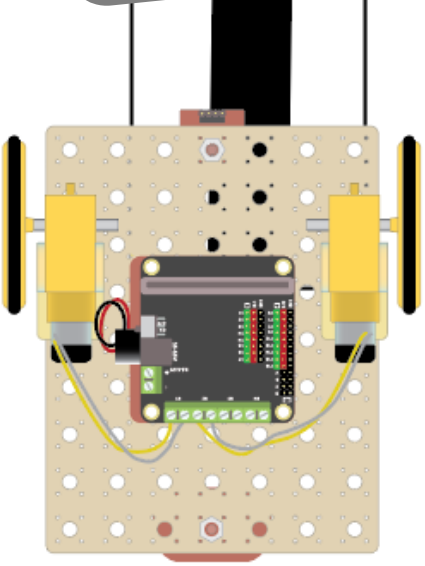
```
309,312
311,314
```

3 Now move the robot so only the left sensor is on the line



```
318,876
305,845
```

4 Now move the robot so only the right sensor is on the line



```
861,309
869,315
```

The numbers you see will be different to these, but you should see a similar pattern.

5 Can you see how these numbers tell you where the line is?

Code the Line Following Sensor 4

Responding to the Readings

Let's now adjust the code so we can work out where the line is.

1 Change your code so that it looks like this

```
forever
    set rightSensor to analog read pin P0
    set leftSensor to analog read pin P1
    set sensorDifference to leftSensor - rightSensor
    if sensorDifference > 50 then
        show leds
    else if sensorDifference < -50 then
        show leds
    else
        show leds
    pause (ms) 200
```

This calculates the difference between the left and right sensor values

If it is darker on the right, **sensorDifference** will be larger than 0.

Show an LED indicating where the black line is (in this case on the right of the robot)

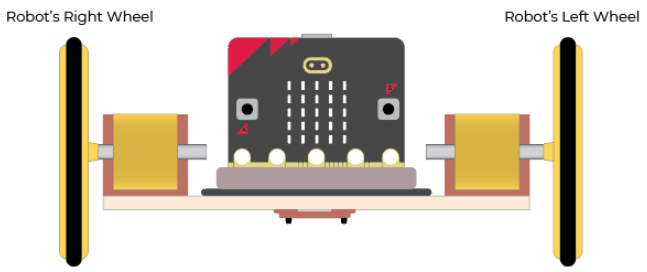
We use 50 and –50 rather than 0 so the robot isn't too sensitive to differences in the sensor values.

If it is darker on the left, **sensorDifference** will be less than 0.

Show an LED indicating where the black line is (in this case on the left of the robot)

Show an LED indicating there is no big difference in the sensor values

This code assumes your robot is set up with left and right like this



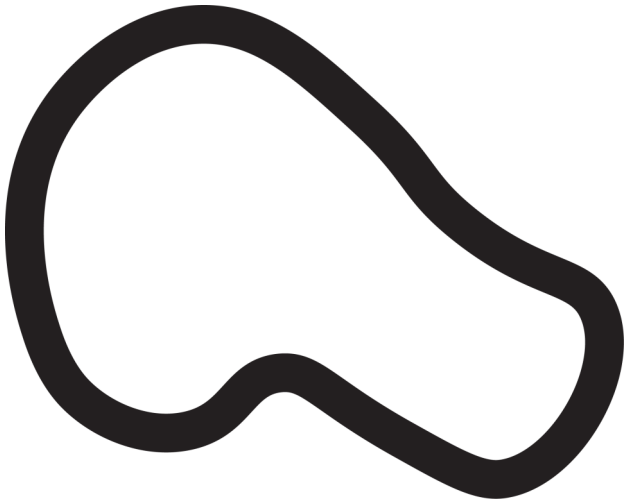
Code the Line Following Sensor 5

Your challenge!

Your code should now show you where the black line is as your robot moves.

Your challenge now is to code your robot so that it moves around the line on the line following mat.

In your code, replace the **show leds** blocks with code that makes your robot adjust its direction so that it stays on the line.



Hints:

- If the robot thinks it is on the black line, which direction should it move?
- If the robot thinks the black line is to the right, which direction should it move?
- If the robot thinks the black line is to the left, which direction should it move?

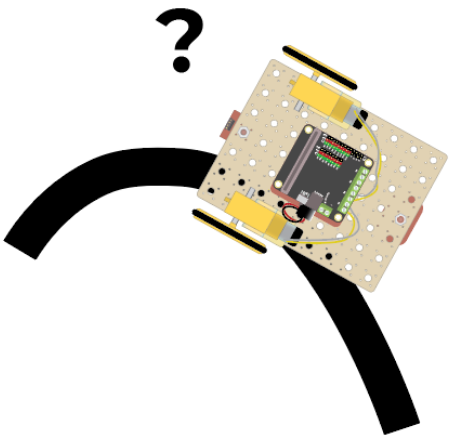
Further hints:

- Start with slow movements of your robot. When your robot starts to follow the line you can try to increase the speed.
- Adjust the 50 and –50 values if you think the sensitivity to the readings isn't quite right.
- The light conditions can affect the ability of the sensors to detect the light. Ensure the line following mat is placed in good light.

Super challenge!

You may notice that when your robot comes completely off the line it gets really confused and doesn't know which way to go!

Can you add some logic to try to help the robot get back on the track?

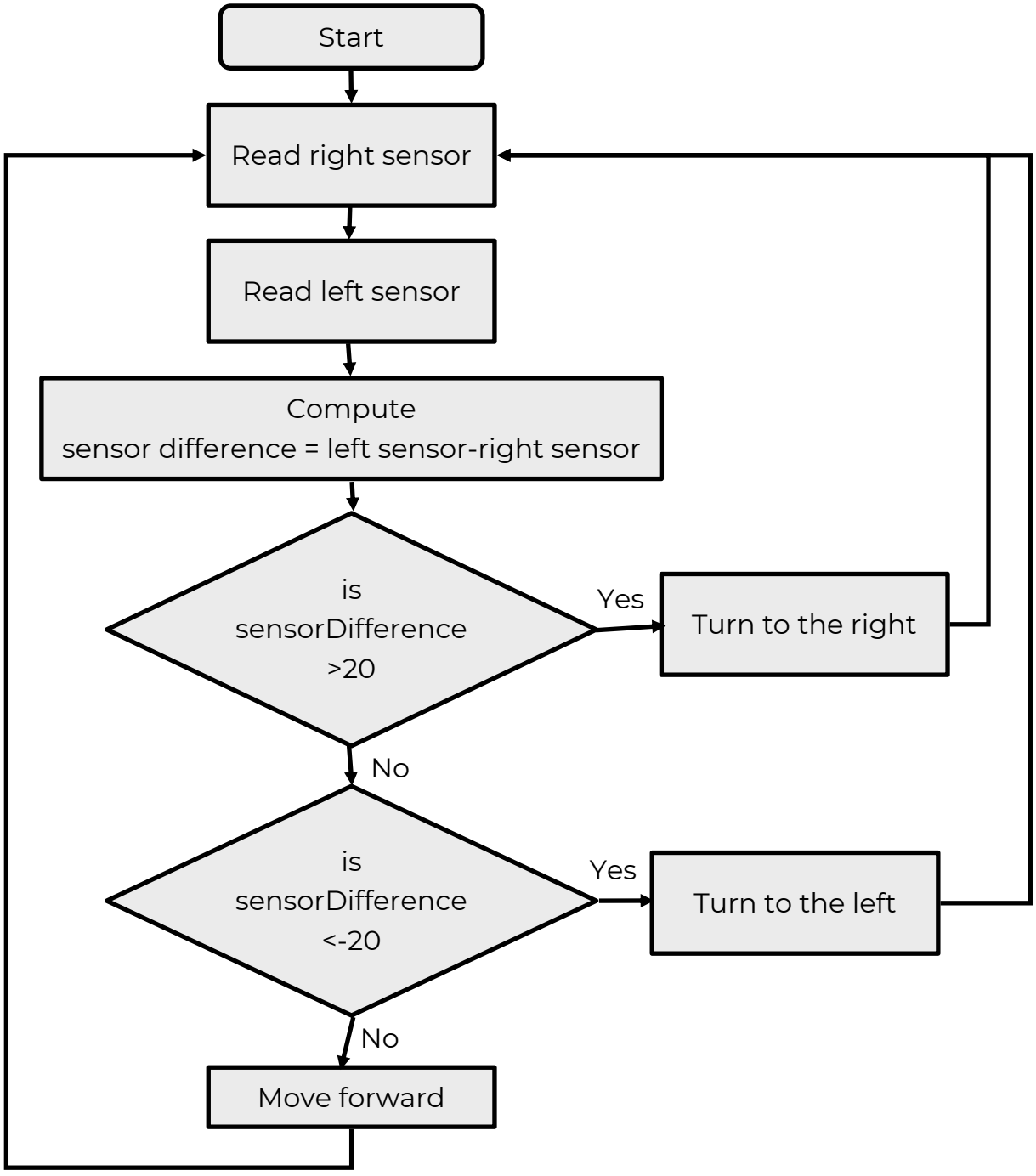


Interesting Fact!

In this project you solved a pretty complicated problem! You gave the Microbit step-by-step instructions on what to do in each situation is encounters. In computer science we use the word **algorithm** to describe such instructions.

Algorithms are used extensively in computing. Algorithms are behind applications that you use to search the internet, or find a route from your home to a friends house. Complex algorithms are behind artificial intelligence (AI) applications that are becoming common these days.

Sometimes computer scientists draw diagrams called **flowcharts** to explain how algorithms work. Here is a flowchart for the line follower robot. Start at the top and follow the logic through. Can you see how it works?



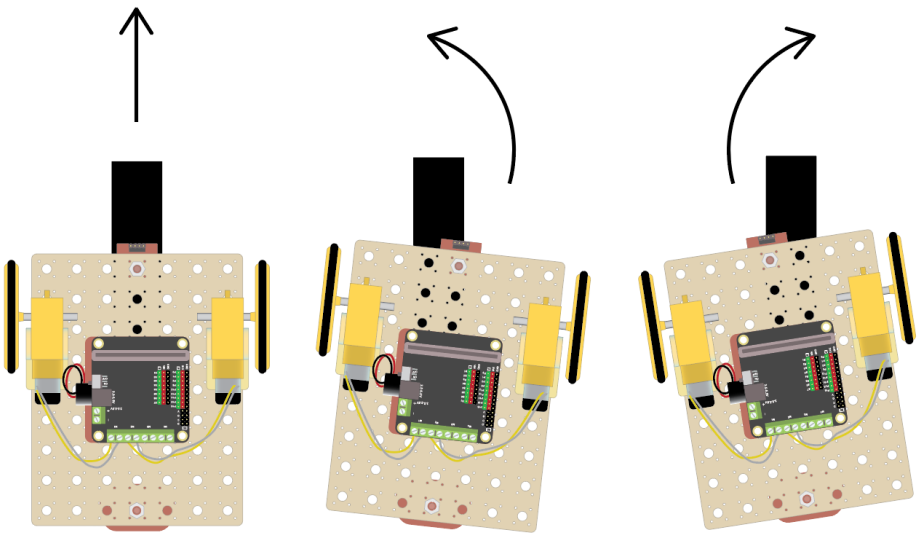
Getting good at creating algorithms is something you can learn to do and it will make you a great programmer and general problem solver!

Solutions

Line Following Robot

You may find that you need slightly different values for your particular robot and light conditions, but this general approach should work.

The idea is that if the robot comes off the line it should turn back towards the line.



Both sensors detecting dark

Action:
Go straight

Left sensor detecting darker than right sensor

Action:
Turn left, back towards the line

Right sensor detecting darker than left sensor

Action:
Turn right, back towards the line

```
forever
    set rightSensor to analog read pin P0
    set leftSensor to analog read pin P1
    set sensorDifference to leftSensor - rightSensor
    if sensorDifference > 20 then
        Motor M1 direction Forward speed 40
        Motor M2 direction Reverse speed 10
    else if sensorDifference < -20 then
        Motor M1 direction Reverse speed 10
        Motor M2 direction Forward speed 40
    else
        Motor M1 direction Forward speed 25
        Motor M2 direction Forward speed 25
```

turn to the right

turn to the left

move forwards

Super Challenge Solution

For the super challenge, you were tasked with adjusting the code to help the robot find its way back on to the line if it comes off. Here is one way to do this.

The general approach is to recognise when both sensors are seeing a white background. This indicates that the robot has come off the line. If this happens we tell the robot to keep turning in the direction it was already going.

```
forever
    set rightSensor to analog read pin P0
    set leftSensor to analog read pin P1
    set sensorDifference to leftSensor - rightSensor
    if rightSensor > 400 and leftSensor > 400 then
        if direction = "R" then
            Motor M1 direction Forward speed 30
            Motor M2 direction Reverse speed 30
        else
            Motor M1 direction Reverse speed 30
            Motor M2 direction Forward speed 30
    else if sensorDifference > 20 then
        set direction to "R"
        Motor M1 direction Forward speed 40
        Motor M2 direction Reverse speed 10
    else if sensorDifference < -20 then
        set direction to "L"
        Motor M1 direction Reverse speed 10
        Motor M2 direction Forward speed 40
    else
        set direction to "F"
        Motor M1 direction Forward speed 25
        Motor M2 direction Forward speed 25
```

If both sensors detecting white

Check the direction we were going before and continue going that way

remember we were going right

remember we were going left

remember we were going forwards