**Pi2Go Programming: Sensors**



**AIM:** After completing this worksheet you should be able to describe the sensors on the Pi2Go robot and use Python to get information from them.

**You Need:** To complete this worksheet you need to have a Pi2Go that is connected to a keyboard, mouse and monitor (see WS1) and to understand how to start IDLE from the Linux Command Line (see WS2).

The Pi2Go robot has four different kinds of sensors:

1. **Ultrasonic:** It has an ultrasonic sensor which uses sound and echoes to measure distances.
2. **Infrared:** It has several infrared sensors that measure how much infrared light is reflected. These can be used both to measure distances (like with the ultrasonic sensor) and to measure colour – for instance a black surface reflects less infra-red light than a white surface.
3. **Light:** The Pi2Go also has several light sensors which detect the intensity of light shining on them.
4. **Switch:** Finally, the Pi2Go has a switch which can detect whether it is pressed or not.

You can use Python to get values from the sensors.

**The Ultrasonic Sensor**



The Ultrasonic Sensor

Locate the ultrasonic sensor on your Pi2Go robot. There is one command you can use to get information from the ultrasonic sensor. This is:

>>> pi2go.getDistance()

pi2go.getDistance(); returns the distance in cm to the nearest reflecting object. A value of zero means it does not detect an object

Type pi2go.getDistance()



**Question 1:** What value do you get?

**Question 2:** Try pointing the sensor at the ceiling or a far wall and getting the distance. What value do you get?

**Question 3:** Now try putting an object close to the sensor.

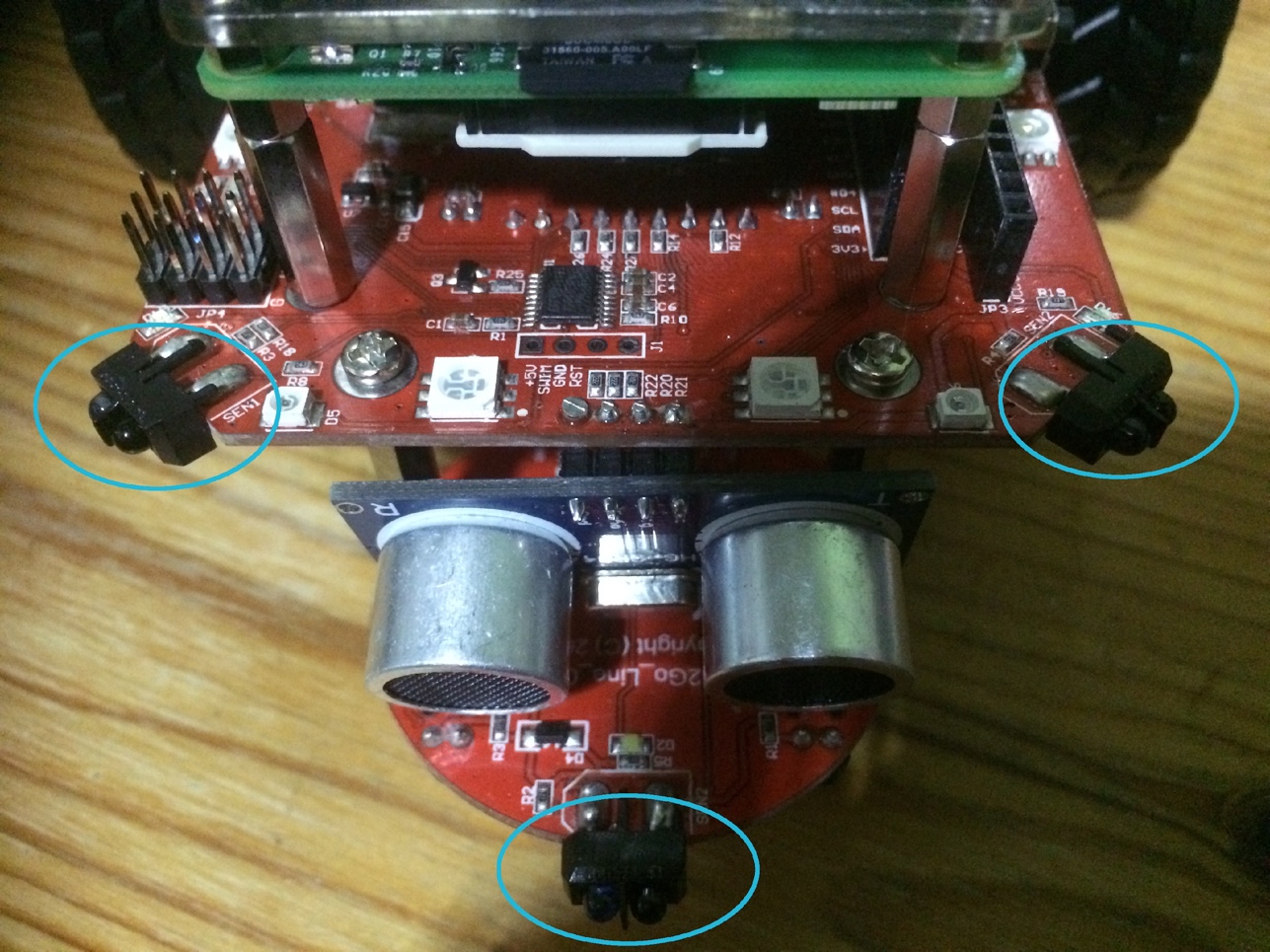


What value do you get?

**Infrared Sensors**

The Pi2Go’s infrared sensors are organised into two groups: sensors for detecting obstacles and sensors for detecting lines.

There are three sensors for detecting obstacles:



The Obstacle Sensors

There are four commands for getting information from the obstacle sensors:

>>> pi2go.irLeft()

returns **True** if the Left IR Obstacle sensor detects an obstacle and **False** otherwise

>>> pi2go.irRight()

returns **True** if the Right IR Obstacle sensor detects an obstacle and **False** otherwise

>>> pi2go.irCentre()

returns **True** if the Centre IR Obstacle sensor detects an obstacle and **False** otherwise

>>> pi2go.irAll()

returns **True** if any of the Obstacle sensors detect an obstacle and **False** otherwise

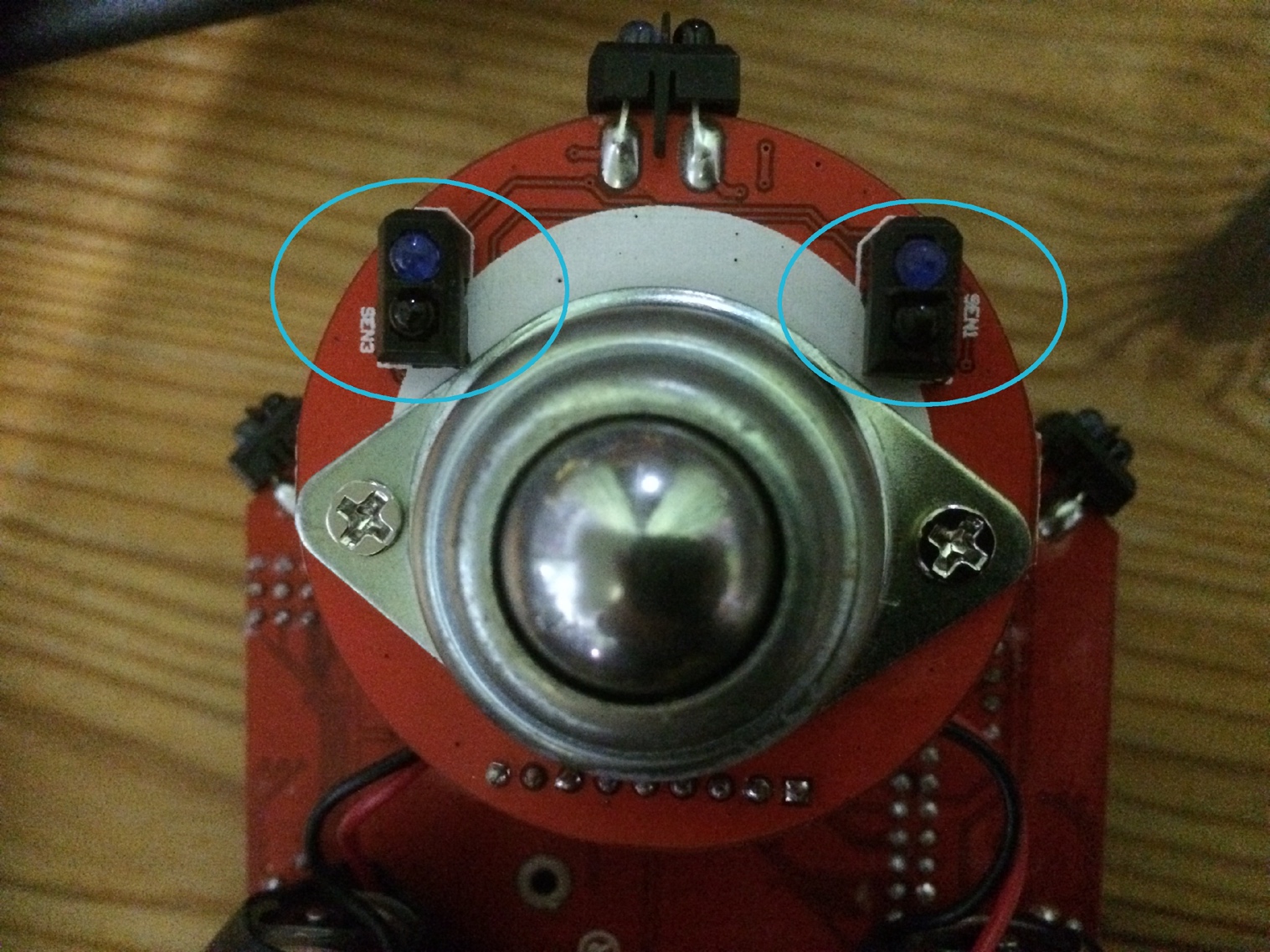
**Question 4:** Use a ruler and the obstacle sensor commands to work out how far away something needs to be to count as an obstacle.

Answer:

**Question 5:** Is this answer the same for all three obstacle sensors? YES/NO

The Pi2Go’s other infrared sensors are used to detect how dark the colour of the surface the robot stands on is. These are referred to as “line sensors” since their primary use is for detecting black lines on white surfaces.

There are two line sensors and you will need to turn your Pi2Go over in order to see them.



The Line Sensors

There are two command for getting information from the line sensors:

>>> pi2go.irLeftLine()

returns state of Left IR Line sensor

>>> pi2go.irRightLine()

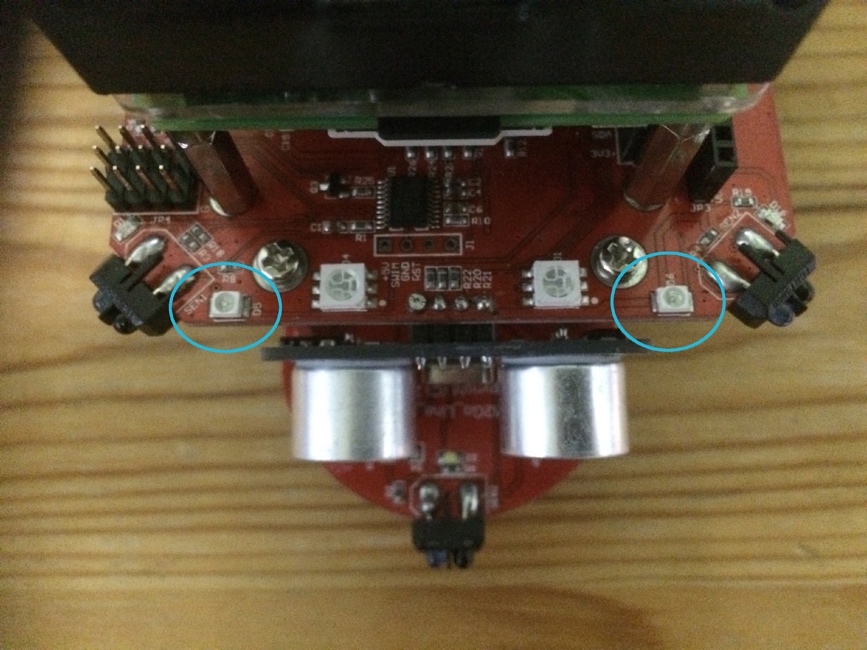
returns state of Right IR Line sensor

**Question 6:** Get a piece of white paper with a black line drawn on it. What do the python commands return if the Pi2Go is over the black line?

**Question 7:** What do they return if the Pi2Go is over the white parts of the paper?

**Light Sensors**

There are four light sensors: two at the front and two at the back.



Front and back light sensors.

There are five commands for getting data from the Pi2Go light sensors.

>>> pi2go.getLight(Sensor)

returns the value 0..1023 for the selected sensor, 0 <= Sensor <= 3

>>> pi2go.getLightFL()

returns the value 0..1023 for Front-Left light sensor

>>> pi2go.getLightFR(0)

returns the value 0..1023 for Front-Right light sensor

>>> pi2go.getLightBL(0)

returns the value 0..1023 for Back-Left light sensor

>>> pi2go.getLightBR(0)

returns the value 0..1023 for Back-Right light sensor

**Note:** Yes it is strange that three of these commands have to take 0 as an argument and one doesn’t. We suspect this is a bug in the pi2go module.

**Question 8:** Check the values returned by one of the sensors when

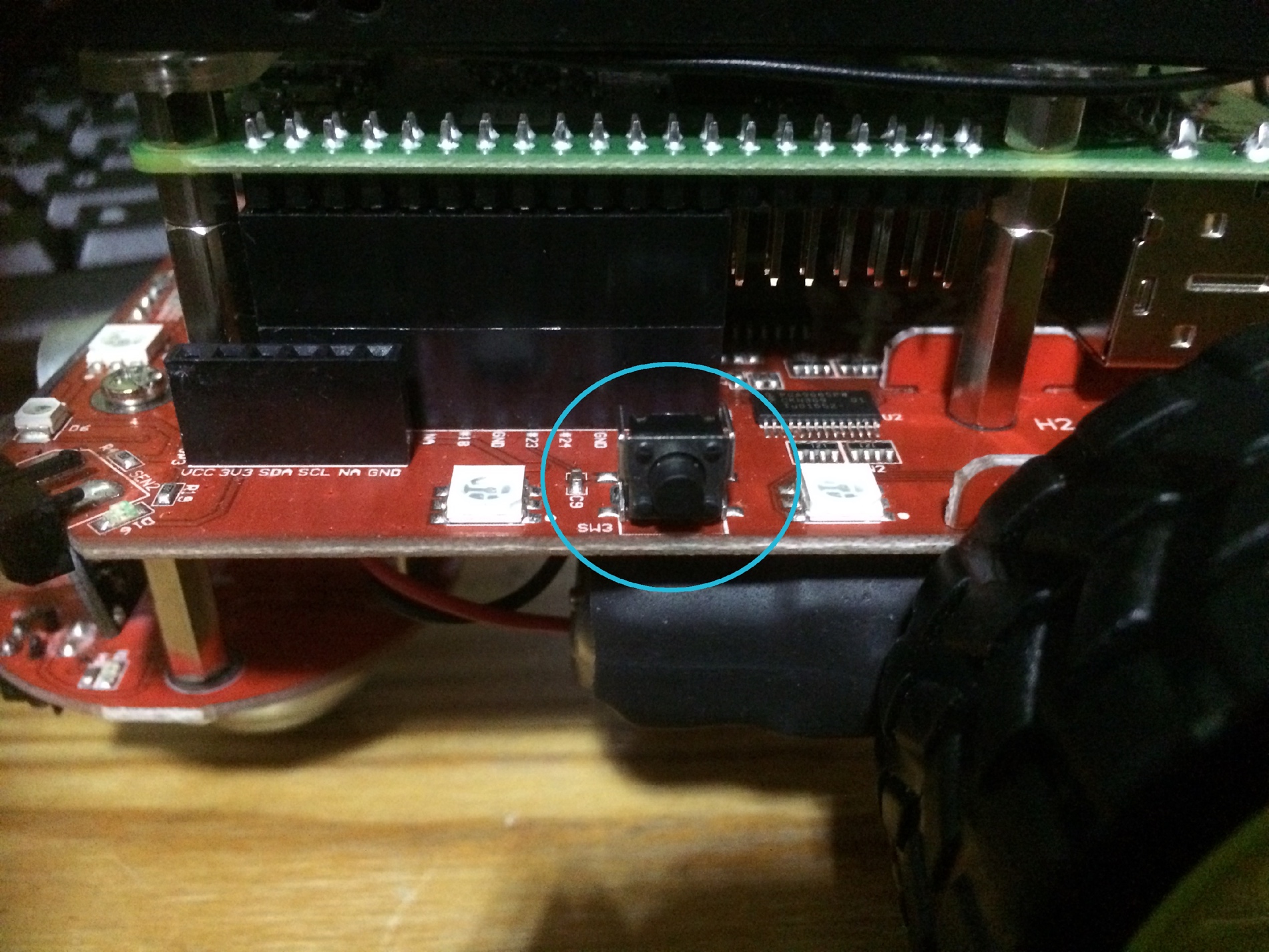
1. you have your hand over it,
2. it is in the natural light of the room and
3. when you shine a torch at it

What value is returned?

Case A Case B Case C

**Switch**

You can find the switch on the side of the Pi2Go



The Pi2Go Switch

There is one command for getting data from the Pi2Go Switch:

>>> pi2go.getSwitch()

returns True when the switch is pressed and False when it isn’t.

**Try using the switch now.**

**Remember:** When you have finished working with the robot, type pi2go.cleanup() at the command line, quit IDLE, then select Shutdown from the Raspberry Pi menu item. Once the robot has shut down, switch it off.



University of Liverpool, 2019

This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-nc-sa/4.0/).