**Virtual Pi2Go Programming: WS11 Sample Answers**

Create a file containing the following program:

import simclient.simrobot as pi2go, time

pi2go.init()

reading1 = pi2go.getDistance()

time.sleep(10)

reading2 = pi2go.getDistance()

if (reading1 < reading2):

print("Object is moving away")

elif (reading2 < reading1):

print("Object is moving closer")

else:

print("Object is not moving")

What do you think this program will do?

**Sample Answer:** The program will take two readings from the ultrasonic (distance) sensor at a 10 second interval. If the first reading is less than the second reading it will print out “Object is moving away”. If the first reading is greater than the second reading it will print out “Object is moving closer”.

How can you test if the program works?

**Sample Answer:** There are three things to test. Firstly I can put a block in front of the robot, not move it, and run the program. It should print out “Object is not moving”. Secondly I can move the block after the program has started running. It should print out either “Object is moving away” or “Object is moving closer” depending upon whether I move the block closer or further away. I should test both these options.

Test the program now. Does it behave as you expect?

**Potential Issues/Trouble Shooting:**

* Cut and paste of program from the work sheet may create syntax errors (particularly to do with the use of “ and indentation inside if statements)
* The simulator initialisation doesn’t complete until after initialisation complete is printed. At this point the first reading is immediately taken. Students who move the block to eagerly may find they get “Object is not moving” - they need to wait for the initialisation complete message before moving the block.
* Students have 10 seconds to move the block after initialisation is complete. This ought to be plenty of time but students do need to be aware of it.

**Exercise:**  Modify the program to make the robot move towards and object that is moving away from it.

**Sample Answer:** Note the use of time.sleep(10) and pi2go.stop()are not necessary to successfully complete the exercise, but they do make a nicer program.

import simclient.simrobot as pi2go, time

pi2go.init()

reading1 = pi2go.getDistance()

time.sleep(10)

reading2 = pi2go.getDistance()

if (reading1 < reading2):

pi2go.forward(10)

time.sleep(10)

pi2go.stop()

**Exercise:**  Modify the program to make the robot “chase” an object by adding a while loop. So long as an object is moving away from it the robot will keep moving towards the object but the robot will stay still if the object is stationary or moving towards it. You can also add in use of the switch to stop the program.

**Sample Answer:** Note I’ve reduced the sleep time in order to make the robot a bit more responsive.

Students may find it useful to use print statements to see the values of reading1 and reading2 in order to debug their programs.

import simclient.simrobot as pi2go, time

pi2go.init()

while (not pi2go.getSwitch()):

reading1 = pi2go.getDistance()

time.sleep(3)

reading2 = pi2go.getDistance()

if (reading1 < reading2):

pi2go.forward(10)

else:

pi2go.stop()

pi2go.stop()



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/).