**Virtual Pi2Go Programming: Variables and Comparisons**



**AIM:** After completing this worksheet you should be able to use variables to store values in Python programs and be able to compare numbers.

**You Need:** To complete this worksheet you need to have a virtual Pi2Go simulation (see WS1), and to be able to use files to store Programs (WS5). You need to know the commands to operate the virtual Pi2Go motors and sensors (WS3 & WS4) and how to use sleep (WS6), if (WS7) and loops (WS8 & WS10) in Python programs.

**If the simulator isn’t already running: Start the Simulator, Select the Pi2Go Simulation and default\_world.xml, then start IDLE (open a *new IDLE window* if you have used IDLE to start the simulator).**

When we program, we frequently want to store a value and reuse it later. For instance, we might want to take two readings from one of the robot’s sensors, one after the other, and compare them.

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")

In this program reading1 and reading2 are *variables.* We use them to store the values of two readings from the distance sensor taken 10 seconds apart and then compare them using < (less than).

**Remember:** elif, this means *else if*. So, the program has three options 1) if reading1 is less than reading2, 2) else if reading2 is less than reading1 3) else - this last option happens if both readings are the same.

**Question 1:** What do you think this program will do?

**Question 2:** How can you test if the program works?

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

**Exercise 1:**  Modify the program to make the robot move towards an object that is moving away from it.

Write your program below:

**More on Comparisons:** As well as checking if one number is less than another we might want to check if they are equal, or not equal. We can use the following **comparison operators** in Python:

|  |  |
| --- | --- |
| a < b | a is strictly less than b |
| a > b | a is strictly greater than b |
| a <= b | a is less than or equal to b |
| a >= b | a is greater than or equal to b |
| a == b | a equals b |
| a != b | a does not equal b |

**Exercise 2:** Write a program that uses == and != to decide whether an object is moving or not. It should print out Object Moving! if the object is moving and Object Stopped! if the object has stopped

**Exercise 3:**  Write a 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 and reverse if the object is moving towards it. You can also add in use of the switch to stop the program.

Write your program below:

**Remember:** When you have finished working with your robot type:

**pi2go.cleanup()**

When want to exit the simulator select the simulator window and type Q.



 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/).