**Pi2Go Programming: Using Logic in Conditions**

**AIM:** After completing this worksheet you should be able to use logic to construct more complex conditions for If statements.

**You Need:** To complete this worksheet you need to have a Pi2Go that is connected to a keyboard, mouse and monitor (see WS1), to understand how to start and stop IDLE from the Linux Command Line (see WS2), and to be able to use files to store Programs (WS5). You also need to know the commands to operates the Pi2Go motors, LEDs and sensors (WS3 & WS4) and how to use while statements in Python (WS7).

**Problem:** You will have noticed that it is very cumbersome to operate your Pi2Go while it is connected to the keyboard, mouse and monitor. Ideally you would have time after your program started running to disconnect everything.

Consider the program below

import pi2go

pi2go.init()

while not (pi2go.getSwitch())

print(“Waiting”)

pi2go.setAllLEDs(4095, 4095, 4095)

pi2go.setAllLEDs(0, 0, 0)

**Question 1:** What do you expect this program to do?

Execute the program. Did it do what you expected? YES/NO.

**Exercise 1:**  Modify the program by adding a second while loop so that the LEDs stay on until the switch is pressed a second time.

**Logic in conditions:** You can use *logic* to make the conditions of your Python loops and if statements more complex. The main *logical operators* are not, and and or.

We can use these to build up logical expressions so, for instance:

|  |  |
| --- | --- |
| pi2go.getSwitch() | returns True if the switch is pressed |
| not (pi2go.getSwitch()) | returns True if the switch is not pressed |
| (pi2go.getSwitch() or pi2go.irCentre()) | returns True if the switch is pressed or there is an obstacle in the centre |
| (pi2go.getSwitch() and pi2go.irCentre()) | returns True if the switch is pressed and there is an obstacle in the centre |

**Exercise 2:** Modify your program again so that if switch is pressed or there is an obstacle then the Pi2Go reverses. Then when the switch is turned on (a second time if it was pressed the first time), the Pi2Go stops moving.

What expression are you using in the condition (if statement)?

You can make conditions more and more complex by composing the logical operators.

**Exercise 3:** Modify your program again so that if switch is pressed and there is not an obstacle then the Pi2Go moves forward. Then when the switch is pressed a second time, Pi2Go stops moving.

What expression are you using in the condition?

**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/).