**Pi2Go Simulator Programming: WS16 Sample Answers and Trouble Shooting**

**WS16**

**Exercise 1:**

import simclient.simrobot as pi2go

import time

def flash\_LEDs():

pi2go.setAllLEDs(500, 500, 500)

time.sleep(1)

pi2go.setAllLEDs(0, 0, 0)

time.sleep(1)

pi2go.setAllLEDs(500, 500, 500)

time.sleep(1)

pi2go.setAllLEDs(0, 0, 0)

pi2go.init()

while (not pi2go.irCentre()):

pi2go.forward(10)

pi2go.stop()

flash\_LEDs()

while (pi2go.irCentre()):

pi2go.spinLeft(10)

time.sleep(5)

pi2go.stop()

flash\_LEDs()

while (not pi2go.irCentre()):

pi2go.forward(10)

pi2go.stop()

flash\_LEDs()

A cleverer answer to Exercise 1 uses a second function as follows:

import simclient.simrobot as pi2go

import time

def flash\_LEDs():

pi2go.setAllLEDs(500, 500, 500)

time.sleep(1)

pi2go.setAllLEDs(0, 0, 0)

time.sleep(1)

pi2go.setAllLEDs(500, 500, 500)

time.sleep(1)

pi2go.setAllLEDs(0, 0, 0)

def while\_no\_obstacle():

while (not pi2go.irCentre()):

pi2go.forward(10)

pi2go.stop();

pi2go.init()

while\_no\_obstacle()

flash\_LEDs()

while (pi2go.irCentre()):

pi2go.spinLeft(10)

time.sleep(5)

pi2go.stop()

flash\_LEDs()

while\_no\_obstacle()

flash\_LEDs()

**Question 1&2:** The turn(side) function turns left or right depending upon its argument. The following program turns the robot right for 10 seconds.

**Exercise 2:**

def turn\_obstacle(side):

if (side == 'right'):

pi2go.spinLeft(10)

while(pi2go.irRight()):

continue

else:

pi2go.spinRight(10)

while(pi2go.irLeft()):

continue

pi2go.stop()

pi2go.init()

turn\_obstacle('left')

**Question 3:** returns the value of the obstacle sensor on the left, right or centre depending upon its argument.

**Exercise 3:**

import simclient.simrobot as pi2go

import time

def obstacle(side):

if (side == 'left'):

return pi2go.irLeft()

elif (side == 'right'):

return pi2go.irRight()

else:

return pi2go.irCentre()

pi2go.init()

print(obstacle('left'))

print(obstacle('centre'))

print(obstacle('left'))

**Exercise 4:**

def opposite(side):

if (side == 'left'):

return 'right'

else:

return 'left'

**Exercise 5:**

import simclient.simrobot as pi2go

import time

def obstacle(side):

if (side == 'left'):

return pi2go.irLeft()

elif (side == 'right'):

return pi2go.irRight()

else:

return pi2go.irCentre()

def turn(side):

if (side == 'left'):

pi2go.spinLeft(10)

else:

pi2go.spinRight(10)

def turn\_until(side):

turn(side)

while(obstacle(opposite(side))):

continue

pi2go.stop();

def opposite(side):

if (side == 'left'):

return 'right'

else:

return 'left'

pi2go.init()

turn\_until('left')



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/).