**Department of Electrical Engineering**

**Faculty Member:**  **Date:**

**Semester: Group:**

# EE381 Robotics

**Lab 1: Python Programming in Linux Ubuntu**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **PLO5-CLO4** | | **PLO9-CLO4** |  |
| **Name** | **Reg. No** | **Analysis of Data in Lab Report** | **Modern Tool Usage** | **Individual and Team Work** | **Total Marks** |
|  |  | **10 Marks** | **5 Marks** | **5 Marks** | **20 Marks** |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

## Introduction

This laboratory exercise is meant to introduce the fundamental aspects of the python programming language which will be very important when learning the Robot Operating System (ROS) for programming robots.

## Objectives

The following are the main objectives of this lab:

* Create variables of different data types in python
* Use arithmetic and logical operations in python
* Implement conditional statements and loops in python
* Create functions and call them in python
* Run the python scripts using the terminal in Linux Ubuntu

## Lab Conduct

* Respect faculty and peers through speech and actions
* The lab faculty will be available to assist the students. In case some aspect of the lab experiment is not understood, the students are advised to seek help from the faculty.
* In the tasks, there are commented lines such as #YOUR CODE STARTS HERE# where you have to provide the code. You must put the code between the #START and #END parts of these commented lines. Do NOT remove the commented lines.
* Use the tab key to provide the indentation in python.
* When you provide the code in the report, keep the font size at 12

**Theory**

Python is an interpreted language which is very popular in the field of robotics. It has an easy-to-learn syntax and is ideal for developing robot prototypes in a short time. To write python scripts (.py files), text editors with syntx highlighting like SublimeText application can be used. Once the code is written, the script is saved and can be executed by using the Linux terminal where the text output is displayed.

The terminal commands are given as:

**cd <directory>** change directory

**cd..** go back to previous directory

**pwd**  print the current directory

**ls**  list the contents of the current directory

**python <script.py>** execute python script

A brief summary of the relevant keywords and functions in python is provided below. (For more details, check the slides for this lab)

**print()** output text on console

**input()** get input from user on console

**range()**  create a sequence of numbers

**len()** gives the number of characters in a string

**if** contains code that executes depending on a logical condition

**else** connects with **if** and **elif**, executes when conditions are not met

**elif** equivalent to **else if**

**while** loops code as long as a condition is met

**for** loops code through a sequence of items in an iterable object

**break** exit loop immediately

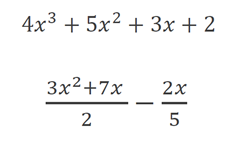
**continue** jump to the next iteration of the loop

**def** used to define a function

For each of the given tasks, execute the script in the Linux terminal. The screenshots must show all relevant output in the terminal.

**Lab Task 1 – Operator Precedence**

Write a program which evaluates the following expressions for when x = 1,2,3,4 and 5.



1. Fill the following table with the answers:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | x = 1 | x = 2 | x = 3 | x = 4 | x = 5 |
| Expression 1 |  |  |  |  |  |
| Expression 2 |  |  |  |  |  |

1. Provide the code and relevant screenshots for both expressions in the indicated regions:

***### EXPRESSION 1 CODE STARTS HERE ###***

*### EXPRESSION 1 CODE ENDS HERE ###*

***### EXPRESSION 2 CODE STARTS HERE ###***

*### EXPRESSION 2 CODE ENDS HERE ###*

***### TASK 1 SCREENSHOT STARTS HERE ###***

*### TASK 1 SCREENSHOT ENDS HERE ###*

**Lab Task 2 – Input Prompt and Type Casting**

Write a program that reads in two integer inputs, then determines and prints if the first is a multiple of the second. To input a variable, use the following syntax:

***variable = input(“prompt\_message”)***

Remember that the above function returns a string which is stored in the variable. You need to explicitly convert the string variable to an integer type.

***### TASK 2 CODE STARTS HERE ###***

*### TASK 2 CODE ENDS HERE ###*

***### TASK 2 SCREENSHOT STARTS HERE ###***

*### TASK 2 SCREENSHOT ENDS HERE ###*

**Lab Task 3 – Selection Statements**

Write a program that prompts the user for two numbers as input. Then, the program must compare the two numbers and print if they are equal or not. If the numbers are not equal, it must also print which number is greater (or lesser) than the other. The syntax for conditional statements is given below:

**if *condition*:**

***statement\_1***

***else:***

***statement\_2***

***### TASK 3 CODE STARTS HERE ###***

*### TASK 3 CODE ENDS HERE ###*

***### TASK 3 SCREENSHOT STARTS HERE ###***

*### TASK 3 SCREENSHOT ENDS HERE ###*

**Lab Task 4 – Relational Operators**

Write a program that prompts the user for two numbers as input. Then, the program must compare the two numbers and print one of the following lines:

* Both numbers are +ve
* Both numbers are -ve
* Both numbers are zero
* At least one number is zero
* One number is +ve and the other number is -ve

***### TASK 4 CODE STARTS HERE ###***

*### TASK 4 CODE ENDS HERE ###*

***### TASK 4 SCREENSHOT STARTS HERE ###***

*### TASK 4 SCREENSHOT ENDS HERE ###*

**Lab Task 5 – WHILE Loop**

Write a program that calculates the factorial of a number. The program must prompt the user for the input number. To calculate the factorial, you need to make use of a WHILE loop. The syntax of the WHILE loop is given as:

**while *condition*:**

***statement\_1***

***statement\_2***

Provide the code and the screenshot showing at least 10 calculations.

***### TASK 5 CODE STARTS HERE ###***

*### TASK 5 CODE ENDS HERE ###*

***### TASK 5 SCREENSHOT STARTS HERE ###***

*### TASK 5 SCREENSHOT ENDS HERE ###*

**Lab Task 6 - Functions**

Write a function that takes 2 integer arguments and returns their product but you must NOT use the product operator (\*).You need to provide the function definition and the function call. (Hint: You can make use of loops in your function.) The function definition syntax is given below:

**def *function\_name*:**

***statement\_1***

***…***

***return output***

***### TASK 6 CODE STARTS HERE ###***

*### TASK 6 CODE ENDS HERE ###*

***### TASK 6 SCREENSHOT STARTS HERE ###***

*### TASK 6 SCREENSHOT ENDS HERE ###*

**Lab Task 7 – FOR Loop**

Write a program that prompts the user for 3 strings variables. The user will input the 3 strings separately at the prompt. The 3 strings must be from the names of your group members. The strings will then be passed to a function as arguments. The function must use a FOR loop to iterate through the all of the characters and print them. Note that each individual character is to be printed on a new line. Provide the code and the screenshot showcasing your work. You need to show both the user inputs and resulting output in the same terminal. The FOR loop syntax is given as:

**for index in *iterable*:**

***statement\_1***

***statement\_2***

***### TASK 7 CODE STARTS HERE ###***

*### TASK 7 CODE ENDS HERE ###*

***### TASK 7 SCREENSHOT STARTS HERE ###***

*### TASK 7 SCREENSHOT ENDS HERE ###*