**Department of Electrical Engineering**

**Faculty Member:** LE Munadi Sial **Date:** 13-Sep-2023

**Semester:** 7th **Group:**

# CS471 Machine Learing

**Lab 1: Introduction to Python**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  | **PLO4 - CLO4** | **PLO4 -CLO4** | **PLO5 -CLO5** | **PLO8 -CLO6** | **PLO9 -CLO7** |
| **Name** | **Reg. No** | **Viva /Quiz / Lab Performance** | **Analysis of data in Lab Report** | **Modern Tool Usage** | **Ethics** | **Individual and Team Work** |
|  |  | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

## Introduction

This laboratory exercise will introduce the fundamental aspects of the Python programming language which is a very popular language and used extensively in the area of Machine Learning.

## Objectives

The following are the main objectives of this lab:

* Write and execute python code in Google Colaboratory (Colab)
* Create and use variables of different data types in python
* Use arithmetic and logical operations in python
* Implement conditional statements in python
* Implement WHILE and FOR loops in python
* Define and call functions in python

## 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/screenshot/plot 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 open-source, interpreted language which is widely used for machine learning tasks in research, academia and industry. It has an easy-to-learn syntax and is ideal for writing programs in a short duration. The python interpreter can be downloaded from the website and installed on the system. By default, the IDLE program is installed. For machine learning, it is recommended to switch to a more powerful IDE such as PyCharm, Spyder and Jupyter etc. For this lab, we will use Google Colab for writing python code. Google Colab is a cloud-based platform that allows you to write python code in your web browser and provides free access to computing resources such as GPUs.

A brief summary of the relevant keywords and functions in python is provided below:

**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 true

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

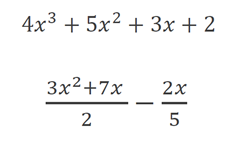
**break** exit loop immediately

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

**def** used to define a function

**Lab Task 1 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [2]**

Write a program which evaluates the following three 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 | 14 | 60 | 164 | 350 | **642** |
| Expression 2 | 4.6 | 12.2 | 22.8 | 36.4 | **53.0** |

1. Provide the code 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 ###*

**Lab Task 2 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

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 using the int() casting. Provide the code and screenshot of the result.

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

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

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

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

**Lab Task 3 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

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 as follows:

**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 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

Write a program that takes two numbers as inputs. Then, the program must compare the two numbers and print appropriately from among the following lines:

* Both numbers are positive
* Both numbers are negative
* Both numbers are zero
* At least one number is zero
* One number is positive and the other number is negative

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

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

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

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

**Lab Task 5 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

Write a program that calculates the factorial of a number. To calculate the factorial, you will need to make use of a *while* loop. The syntax of the while loop is given as follows:

**while *condition*:**

***statement\_1***

***statement\_2***

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

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

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

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

**Lab Task 6 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

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

**def *function\_name*:**

***statement\_1***

***statement\_2***

***…***

***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 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

Write a program that prompts the user for 3 strings variables. The user will input the strings separately at the prompt, e.g. “TRI”, “GONO” and “METRY”. The strings will then be passed to a function as arguments. The function must use a *for* loop to iterate through the characters and print each character on a new line. The function must also print the total number of characters in the final string. For this, you can use the len() function. Note that the “TRIGONOMETRY” string is just an example and you need to use your own string for the submission. You also need to take screenshot of this task showing the entire output. The for loop syntax is given as follows:

**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 ###*

**Lab Task 8 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ [1]**

Write a program that generates the following number sequences and print the output. You can use the range() function for this task. Use a loop to invoke the range function iteratively.

1, 2, 3… 20

2, 4, 6… 40

3, 6, 9… 60

4, 8, 12 … 80

…

10, 20, 30… 200

***### TASK 8 CODE STARTS HERE ###***

*### TASK 8 CODE ENDS HERE ###*

***### TASK 8 SCREENSHOT STARTS HERE ###***

*### TASK 8 SCREENSHOT ENDS HERE ###*