**Date: 2024-02-11**

**Assignment 1**

This assignment consists of the programming questions related to the topics of week 1 and week 2. The main topics of questions are: Python Basics, Operators, and Conditional Statements.

All the students are required to follow the format of the program as specified in the guideline below.

1. All the programs should have initial **doc string** comment (‘’’ description of program‘’’) mentioning what your program will do.
2. Try to maintain single/multi-line comments in the place where needed to make the program understandable.
3. Maintain proper indention and newline spaces to increase the readability of the program.
4. The deliverable are 2 type of files (a single word file and multiple python program files):
   1. Separate python program files with **.py** extension (e.g. program\_name.py). Provide a relevant name to your program file on the basis of functionality of the program.
   2. A word file describing the working of all the programs according to their number. The details required in this is the description of program, screenshot of the testing (input given and output obtained in the execution environment such as IDLE or Command prompt or terminal whichever you prefer.). It is preferred that you work with multiple inputs and outputs.

**Questions**

1. Create a diagram to show how python works. The diagram should show the components such as input, processing and output.
2. Write a program to input 2 numbers from the users and display the output of below mentioned operations in a proper format.
3. Addition
4. Subtraction
5. Multiplication
6. Division
7. Modulo division
8. Floor division
9. Write a program to take a number input from the user and display whether the number is even or odd.
10. Write a program to take a number input from the user and display the result of some mathematical calculations as mentioned below.
11. Square of the number
12. Square root of the number
13. Exponent value with the number
14. Log Base 10 of the number
15. Calculate the power 3, 4 and 5 of the number.
16. Solve the below mentioned expressions in a python program. Feel free to take input of the required variables to solve the expressions.
17. a2 + 2ab + b2
18. a5 + 2abc + b3 + c4
19. a7 + 5a3b2c6 + b7
20. Write a program to input the number of 5 subjects from the user, calculate the average, total, percentage and division of the students on the basis of specifications mentioned below.
21. If the percentage value is 80 or above, the user obtains distinction.
22. If the percentage is above 60, the user obtains first division.
23. If the percentage is above 50, the user obtains second division.
24. If the percentage is above 45, the user obtains third division.
25. If the percentage is below 45, the user obtains fail division.
26. Write a program to display the prime number between 2 numbers input by the users. Also print the sum of all the prime numbers. [Hint: Prime numbers are the one which are either divisible by 1 or themselves. 3, 5, 7, 11, etc are some of the examples.]
27. Write a program which will find all such numbers that are divisible by 7 but are not a multiple of 5,

between 2000 and 3200 (both included). The numbers should be printed on the output screen. Also try the same program by replacing 2000 and 3200 by taking input for them from the users.

1. Write a program to find the factorial of any number taken as an input from the user. Try to validate the user input whether it is a number or not and then only perform the operation. In case of other than number as an input, display an error as “Not a number.”. [Hint: few available functions to identify the input is a number or not are ‘**isdigit()**, **isnumeric()**, etc.]
2. Write a program to find the sum of odd and even numbers input from the user. The program should take an input continuously until the user wants to exit the program. The program should be menu driven where the user should be asked whether they want to continue with the program or not.
3. Write a program to create a number guessing game for the user. The program should ask the user to input a number. The program specifications are as mentioned below.
4. The program should generate a random number for the answer.
5. The program should prompt the user for a number input.
6. The program should provide the feedback to the user after each guesses (e.g. “Too high”, “Too low” or “Correct number”).
7. The program should check the user input for 5 times and allow the users to guess for at most 5 times if their input don’t match the answer number.
8. If the user is not able to guess the answer within 5 times, the program should display “Game Over” message and exit.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***End of Assignment 1** \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*