**Experiment 5: Title of the Laboratory Exercise: Logic operations and decision making**

1. Introduction and Purpose of Experiment

   Python provides number of control flow instructions/statements to control the flow of program execution conditionally. By solving the problems, students will be able to apply conditional control statements to control the program execution.

2. Aim and Objectives

   Aim

   - To develop programs involving branching using appropriate control statements in Python

   Objectives

   At the end of this lab, the student will be able to

   - Apply conditional control statements such as *if-else* and nested *if-else* to express decisions

3. Experimental Procedure

   i. Analyse the problem statement
   ii. Design an algorithm for the given problem statement and implement the algorithm in Python language
   iii. Execute the Python program
   iv. Test the implemented program
   v. Document the Results
   vi. Analyse and discuss the outcomes of the experiment

**Problems:**

1. Write a Python program to check whether the given number is zero, positive or negative.

2. Write a Python program to check whether the given year is leap year or not.

3. Write a Python program to display the roots of a quadratic equation.

4. Compute the Body Mass Index.

Instructions: a. Use the nested if statements to write a program to compute the BMI

BMI: BMI is a measure of health based on weight. It can be calculated by taking your weight in kilograms and dividing it by the square of your height in meters. The interpretation of BMI for people 16 years and older is as follows:

| **BMI** | **Interpretation** |
|---|---|
| Below 18.5 | Underweight |
| 18.5- 24.9 | Normal |
| 25.0-29.9 | Overweight |
| Above 30.0 | Obese |

Write a program that prompts the user to enter a weight in pounds and height in inches and then display the BMI. Note that one pound is 0.45359237 kilograms and one inch is 0.0254 meters.

**Sample Outcome:**

Enter weight in pounds: 146

Enter height in inches: 70

BMI is 20.95

Normal

5. Game: Heads or tails

Write a program that lets the user guess whether a flipped coin displays the head or the tail. For this, the program randomly generates a number 0 or 1, which represents head or tail. The program prompts the user to enter a guess and reports whether the guess is correct or incorrect.

6. Game: Scissor, Rock, and Paper

Write a program that plays the popular scissor-rock-paper game. Instruction: A scissor can cut the paper, a rock can knock a scissor, and a paper can wrap a rock. For this, the program randomly generates a number 0, 1, or 2 representing scissor, rock, and paper. The program prompts the user to enter a number 0, 1, or 2 and displays a message indicating whether the user or the computer wins, loses, or draws.

**Here, are sample runs:**

Scissor (0), rock (1), paper (2): 1

The computer is Scissor. You are the rock. You won

Scissor (0), rock (1), paper (2): 2

The computer is paper. You are paper too. It is a draw