Lab 1: Introduction to Python

Python is a widely used high-level, general-purpose, interpreted, dynamic programming language. Its design philosophy emphasizes code readability, and its syntax allows programmers to express concepts in fewer lines of code than possible in languages such as C++ or Java.

Lab Work:

1)WAP to check if an input number is odd or even

2)WAP to input the percentage and display the division

>=80 → Distinction

>=65 → First Division

>=55 → Second Division

>=40 → Third Division

<40 → Fail

3)WAP to calculate sum, diff, product and quotient between two input numbers using a single function.

4)WAP to display prime numbers from 1 to 100

5)WAP to enter the marks of 10 students and display it.

6)WAP to calculate the factorial of an input number.

7)WAP to ask for a sentence and count the number of words.

8)WAP to sort the list {5, 4, 11, 13, 51}

9)WAP program to sum all the items in a list.

10)WAP program to get the largest number from a list.

11)WAP to ask for a sentence and calculate the frequency of characters in the sentences.

12)WAP to find the sum of all items in a dictionary

Input: {'a': 100, 'b':200, 'c':300}

Output: 600

Input: {'x': 25, 'y':18, 'z':45}
Output: 88

13) You are given a string and your task is to *swap cases*. In other words, convert all lowercase letters to uppercase letters and vice versa.

14) Write a Python program to create a class representing a Circle. Include methods to calculate its area and perimeter.

15) Write a Python program to create a person class. Include attributes like name, country and date of birth. Implement a method to determine the person's age.

16) Define a class `Vehicle` with attributes `make` and `model`, and a method `drive()` which prints "Driving the [make] [model]". Then, create a subclass `Car` that inherits from `Vehicle` and overrides the `drive()` method to print "Driving the [make] [model] car".

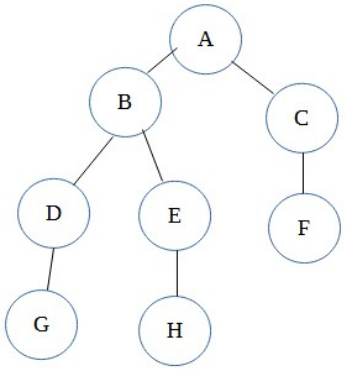
17) Create a class `BankAccount` with private attributes `balance` and `account_number`. Implement methods `deposit()` and `withdraw()` to modify the `balance`. Ensure that the `balance` cannot be accessed directly from outside the class.

18) Implement a class `Shape` with a method `area()` which returns 0. Then, create subclasses `Rectangle` and `Circle`. Overload the `area()` method in both subclasses to calculate and return the area of a rectangle and a circle respectively.

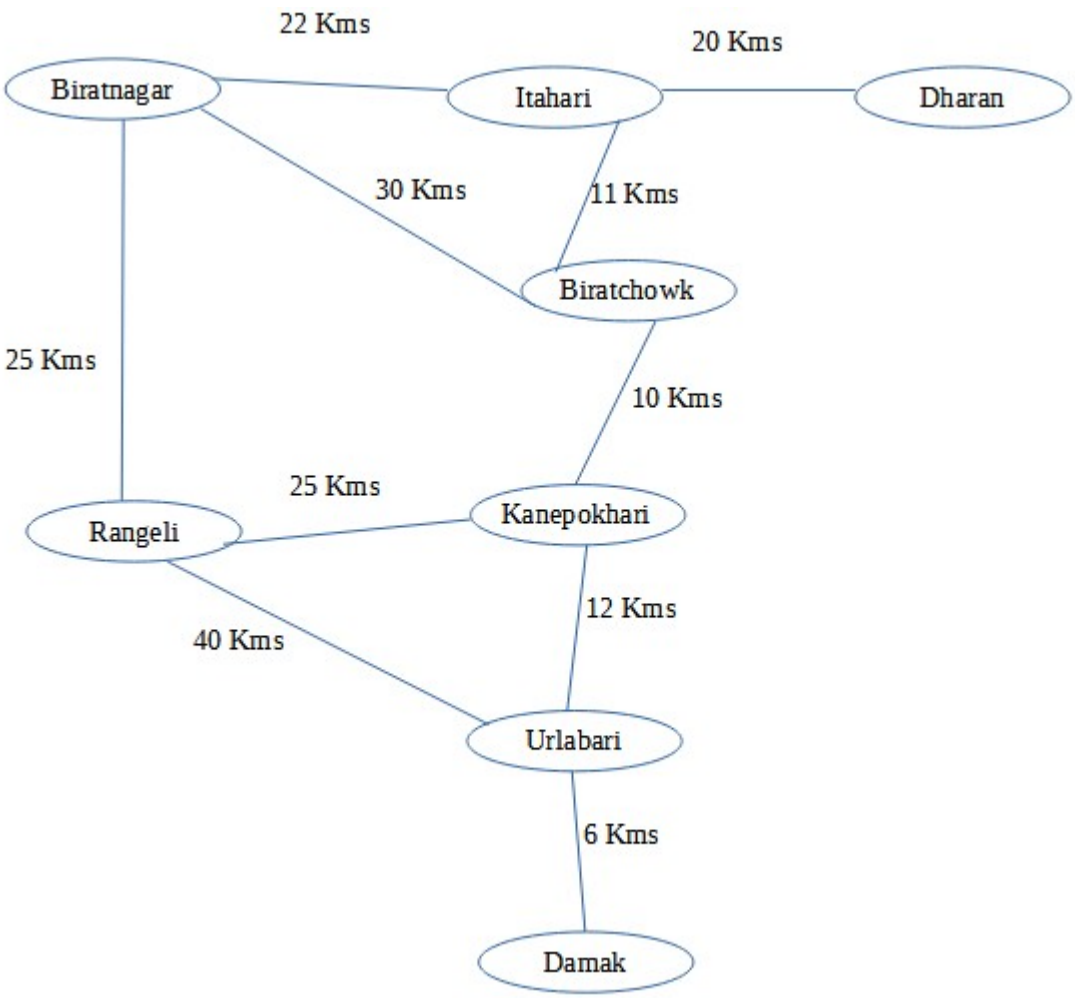
19) Define classes `Engine`, `Wheel`, and `Car`. `Engine` and `Wheel` classes have attributes `type` and methods `start()` and `stop()`. The `Car` class should have instances of `Engine` and `Wheel` classes as attributes. Implement a method `start_car()` in the `Car` class which starts the engine and prints "Car started".

20) WAP to represent the following graphs using a dictionary.

a)



b)


22 Kms
20 Kms
Biratnagar
Itahari
Dharan
30 Kms
11 Kms
Biratchowk
25 Kms
10 Kms
25 Kms
Rangeli
Kanepokhari
12 Kms
40 Kms
Urlabari
6 Kms
Damak