CLO-04:	**Solve problems using Graph data structures and its associated algorithms involving real life relationships between different types of data.**

Assignment Submission Guidelines:
  ➢ Submit your assignment only on google classroom within deadline.
  ➢ Late submissions of upto 15 mins are allowed with a penalty of 20% on google classroom.
  ➢ Add a comment section atop of your code files, mentioning your name roll number etc, and a bit about what you did in this code. Follow the detailed guidelines shared by the TA.
  ➢ Name/rename your submission files as "Section-roll#-Assig#-filenmae.cpp", e.g., "A-20i0899-A2-main.cpp".
  ➢ Zero marks will awarded in following cases
      o Code with build errors
      o Submitting assignment through email, slate, or any other way other than the google classroom
      o Submitting after the late deadline

Finding shortest path between any two cities in Pakistan

You are being provided a map of Pakistan showing roads between different cities of the Pakistan. You can envision this as an undirected non-weighted graph. You need to convert this into a weighted gaph. To convert this into a weighted graph, you have two options.

  • Either you find the length (kms) of roads, or

  • you find the travel time (mins) on these roads.

For any of the above options, you need to use google maps to find the weights ONLY between the directly connected cities.
As you have to submit this activity as well, so do this neatly either directly on your computer or on the paper by first printing the map on the paper and then writing weights on it. Submit scanned copy if you did this work on paper.

Next step is to convert your logical graph into a text file, so that the program should be able to read in all the details of the graph. Internally, all the cities will be treated as numbered vertices, but user will input

only the city names, and the output should also mention the city names. For this, you need to find out the way to assign each city a unique ID number starting from zero and onwards.

Now you need to represent your grpah with a text file. The file pattern should be:

NumberOfVertices

Vertice#   ConnectedTo   Weight   ConnectedTo   Weight   -999

Vertice#   ConnectedTo   Weight   ConnectedTo   Weight   -999

. . .

. . .

Create a class WeightedGraph with functions to
- create the graph (using Adjacency List or Matrix)
- find shortest path between any two cities in the graph using Dijkstra or Floyd-Warshal Algo

If you select Flyod-Warshall, then you just need to run this algo once, and you have the shortest distance info between any two cities, whereas if you select the Dijkstra, you have to run the algo, every time a user inquires for a shortest distance.

The program should present a menu to the user asking for city names to find the shortest path between them

From City name

To City name

OUTPUT:

The program should print

the shortest distance in Kms or time taken from source to destination

all the city names coming in the path from source to destination

Submission:
1. Code files
2. Text file for the graph
3. Weighted Map