Assignment 3: Fun with Graphs

For this assignment, you are to work in groups of two or three.

1. (5 marks) Write a well-documented program in C that reads in graphs, which are in the format specified below, into an *adjacency matrix* data structure. **To make sure you have built the data structure correctly, output the data structure in matrix form.**   Please make sure your program can read graphs of various sizes. An example of a graph in the specified format is provided for you (testGraph.txt).

    Graph Format:
    - The first line contains the number of vertices in the graph. You may assume that if there are *n* vertices, they are uniquely identified by 0, 1, 2,...,n-1.
    - Each of the subsequent lines contains information about an **undirected** edge, which is characterized by 3 numbers: *id of source vertex*, *id of target vertex*, *weight of edge*. Since the edges are undirected, if there is an edge listed from a to b, then your adjacency matrix should include the edge (a,b) as well as (b,a)

2. (5 marks) Add to the program you wrote in #1 to implement the "All-Pairs Shortest Paths" algorithm given in the slides. Your code should output the intermediate graphs as well as final graph (in matrix form).

3. (5 marks) Add to the program you wrote in #2 to implement a function named *centralNode* that returns the *id* of the node whose largest of all the shortest paths to any other node is as small as possible. If there is a **tie** amongst the nodes, the function returns any one of them. If the graph is **not connected**, then this function returns **-1**.

4. (5 marks) Create 3 weighted graphs, which have at least 6 vertices and 10 edges, to <u>test</u> the program's functionality. *(Do not use the provided sample graph. Consider all possible scenarios. Does it produce correct results for connected graphs? Does it produce correct results for graphs which are not connected?)* Your report should include a graphical representation of each of your input graphs as well as the output obtained from running the program. Along with your C program and report, your folder submission should include the corresponding text files for the input graphs used.