ID: 22141026

Task 1

(A)

To solve this task I made a directed weighted graph using adjacency matrix. Initializing matrix with zeros, 2D square array was used to make the matrix.

(B)

Here, same as the above task but here I needed to use adjacency list to make directed weighted graph.

2

I traversed through the whole city using BFS. Starting from root which is city 1; we traverse each node of a level before going to next one.

3

Here, to traverse the city I needed to use DFS. I initially started at the root which was city 1 and before backtracking I traversed whole.

4

I used two list to store visited and stack as to True or False list to find out a graph has a cycle.

5

Here, I used BFS traversal to find the shortest path from root for a given destination. A Queue and a list of visited cities helps to count patway.

Here, I used DFS traversal for maximum number of diamonds count. The total number of nodes containing 'D' - from the root to all the available nodes to a path counted.