Graph

Important Questions:

- Create a class Graph and implement the following functions/methods in that class:

    - addNewEdge(source, destination, distance)
    - printAdjacencyList()
    - bfsTraversal()

    [Follow here: https://www.geeksforgeeks.org/breadth-first-search-or-bfs-for-a-graph/]

    - dfsTraversal()

    [Follow here: https://www.geeksforgeeks.org/depth-first-search-or-dfs-for-a-graph/]

[For help: https://1drv.ms/t/s!AqTOHFO77CqEiRua06v1PATyiFg5]

- Detect cycle in a Directed graph using BFS algo and do the same using DFS algo

[Follow here: https://www.geeksforgeeks.org/detect-cycle-in-a-graph/]

- Detect cycle in a Undirected graph using BFS algo and do the same using DFS algo

[Follow here: https://www.geeksforgeeks.org/detect-cycle-undirected-graph/]

- Write a method to find the shortest path between two nodes using the bfs algorithm.

[Follow here: https://www.geeksforgeeks.org/shortest-path-unweighted-graph/]

- Write a method to find the shortest path between two nodes using Dijkstra's algorithm.

[Follow here: https://www.geeksforgeeks.org/dijkstras-shortest-path-algorithm-greedy-algo-7/]

- Minimum steps to reach target by a Knight

[Follow here: https://www.geeksforgeeks.org/minimum-steps-reach-target-knight/]

- Minimum number of jumps to reach end of given array

[Follow here: https://www.geeksforgeeks.org/minimum-number-of-jumps-to-reach-end-of-a-given-array/]

- Find the number of Islands

[Follow here: https://www.geeksforgeeks.org/find-number-of-islands/]

- Find bridge in a graph

[Follow here: https://www.geeksforgeeks.org/bridge-in-a-graph/]

- Implement Topological sorting algorithm

[Follow here: https://www.geeksforgeeks.org/topological-sorting/]

- Given a sorted Dictionary of an Alien Language, find order of characters

[Follow here: https://www.geeksforgeeks.org/given-sorted-dictionary-find-precedence-characters/]

- Flood Fill Algorithm

[Follow here: https://www.geeksforgeeks.org/flood-fill-algorithm-implement-fill-paint/]

- Rat in a Maze

[Follow here: https://www.geeksforgeeks.org/rat-in-a-maze-backtracking-2/]

- N-Queen Problem

[Follow here: https://www.geeksforgeeks.org/n-queen-problem-backtracking-3/]

- What is MST(Minimum Spanning Tree) ?

[Follow here: https://www.hackerearth.com/practice/algorithms/graphs/minimum-spanning-tree/tutorial/]

- Implement Kruksal's Algorithm

[Follow here: https://www.geeksforgeeks.org/kruskals-minimum-spanning-tree-algorithm-greedy-algo-2/]

- Implement Prim's Algorithm

[Follow here: https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/]

- Total no. of Spanning tree in a graph

[Follow here: https://www.geeksforgeeks.org/total-number-spanning-trees-graph/]

- Minimum Product Spanning Tree

[Follow here: https://www.geeksforgeeks.org/minimum-product-spanning-tree/]

- Implement Bellman Ford Algorithm

[Follow here: https://www.geeksforgeeks.org/bellman-ford-algorithm-dp-23/]

- Implement Floyd warshall Algorithm

[Follow here: https://www.geeksforgeeks.org/floyd-warshall-algorithm-dp-16/]

- Travelling Salesman Problem

[Follow here: https://www.geeksforgeeks.org/traveling-salesman-problem-tsp-implementation/]

- Graph Colouring Problem

[Follow here: https://www.geeksforgeeks.org/graph-coloring-set-2-greedy-algorithm/]

- Snake and Ladders Problem

[Follow here: https://www.geeksforgeeks.org/snake-ladder-problem-2/]

- Count Strongly connected Components (Kosaraju Algo)

[Follow here: https://www.geeksforgeeks.org/strongly-connected-components/]

- Check whether a graph is Bipartite or Not

[Follow here: https://www.geeksforgeeks.org/bipartite-graph/]

- Clone a graph

[Follow here: https://www.geeksforgeeks.org/clone-an-undirected-graph/]

- Detect Negative cycle in a graph

[Follow here: https://www.geeksforgeeks.org/detect-negative-cycle-graph-bellman-ford/]

- Longest path in a Directed Acyclic Graph

[Follow here: https://www.geeksforgeeks.org/longest-path-directed-acyclic-graph-set-2/]

- Minimum cost to connect all cities

[Follow here: https://www.geeksforgeeks.org/minimum-cost-connect-cities/]

- Find if there is a path of more than k length from a source

[Follow here: https://www.geeksforgeeks.org/find-if-there-is-a-path-of-more-than-k-length-from-a-source/]

- M-Colouring Problem

[Follow here: https://www.geeksforgeeks.org/m-coloring-problem-backtracking-5/]

- Hamiltonian Cycle

[Follow here: https://www.geeksforgeeks.org/hamiltonian-cycle-backtracking-6/]

- Permutation of numbers such that sum of 2 consecutive numbers is a perfect square

[Follow here: https://www.geeksforgeeks.org/permutation-numbers-sum-two-consecutive-numbers-perfect-square/]

- Minimum edges to reverse o make path from source to destination

[Follow here: https://www.geeksforgeeks.org/minimum-edges-reverse-make-path-source-destination/]

- Paths to travel each nodes using each edge(Seven Bridges)

[Follow here: https://www.geeksforgeeks.org/paths-travel-nodes-using-edgeseven-bridges-konigsberg/]

- Kth heaviest adjacent node in a graph where each vertex has weight

[Follow here: https://www.geeksforgeeks.org/kth-adjacent-node-graph-vertex-weight/]

- Ford-Fulkerson Algorithm for maximum flow problem

[Follow here: https://www.geeksforgeeks.org/ford-fulkerson-algorithm-for-maximum-flow-problem/]

- Vertex Cover Problem

[Follow here: https://www.geeksforgeeks.org/vertex-cover-problem-set-1-introduction-approximate-algorithm-2/]

- Chinese Postman or Route Inspection

[Follow here: https://www.geeksforgeeks.org/chinese-postman-route-inspection-set-1-introduction/]

- Number of Triangles in a Directed and Undirected Graph

[Follow here: https://www.geeksforgeeks.org/number-of-triangles-in-directed-and-undirected-graphs/]

- Minimise the cashflow mong a given set of friends who have borrowed money from each other

[Follow here: https://www.geeksforgeeks.org/minimize-cash-flow-among-given-set-friends-borrowed-money/]

- Two Clique Problem

[Follow here: https://www.geeksforgeeks.org/two-clique-problem-check-graph-can-divided-two-cliques/]