United International University (UIU)

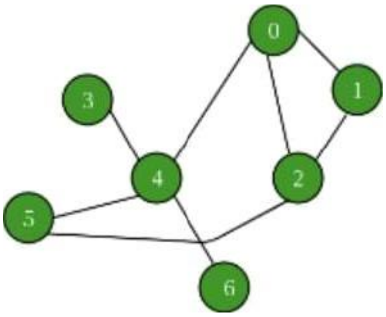
Dept. of Computer Science & Engineering (CSE)

CSE 2216: Data Structure and Algorithms 1 Lab

Assignment 2

Problem 1: You are given an undirected graph G(V, E) with N vertires and M edges. You need to find the minimum number of edges between a given pair of vertices (u, v) using ***Breadth First Search*** **(BFS)**.

For example, the minimum number of edges between vertices (1,5) in the following graph is 2 since (1, 2) and (2, 5) are the only edges resulting in the shortest path between 1 and 5.



Input	Ou¢put
V=9 0 1 0 7 1 7 1 2 2 3 2 5 2 8 3 4 3 5 4 5 5 6 6 7 7 8 findMinNumberEdges(0, 5) findMinNumberEdges(3, 8) findM inNumberEdges(2, 6)	Min number of edges between (0, 5): 3 Min number of edges between (3, 8): 2 Mín number of edges between (2, 6): 2

<u>Problem 2</u>: Mr. Lee has to travel to various branches abroad to assist his company. But he has a problem. The cost would be really high as all branches he has to visit are in foreign countries. He wants to visit every location only one time and return home with the lowest expense. Help this company-caring man calculate the lowest expense using ***Depth First Search*** **(DFS)**.

[Input format]

N, the number of branches (including office) to visit. At this moment, the No. 1 office is regarded as his company (Departure point). Costs are given to move cities in which branches are located from the second line to N number lines. i.e. jth column of ith row is the cost to move from ith city to jth city. If it is impossible to move between two cities, it is given as zero.

[Output format]

Output the minimum cost used to depart from his office, visit all branches, and then return to his home.

<table>
<tr><th>Input</th><th><b>Output</b></th></tr>
<tr><td>Enter N: 5
<table>
<tr><td>0</td><td>14</td><td>4</td><td>10</td><td>20</td></tr>
<tr><td>14</td><td>0</td><td>7</td><td>8</td><td>7</td></tr>
<tr><td>4</td><td>5</td><td>0</td><td>7</td><td>16</td></tr>
<tr><td>11</td><td>7</td><td>9</td><td>0</td><td>2</td></tr>
<tr><td>18</td><td>7</td><td>17</td><td>4</td><td>0</td></tr>
</table></td><td>Minimum cost: 30</td></tr>
<tr><td>Enter N: 5
<table>
<tr><td>9</td><td>9</td><td>2</td><td>9</td><td>S</td></tr>
<tr><td>6</td><td>3</td><td>5</td><td>1</td><td>5</td></tr>
<tr><td>1</td><td>8</td><td>3</td><td>3</td><td>3</td></tr>
<tr><td>6</td><td>0</td><td>9</td><td>6</td><td>8</td></tr>
<tr><td>6</td><td>6</td><td>9</td><td>4</td><td>8</td></tr>
</table></td><td>Minimum cost: 18</td></tr>
</table>

Instructions:

1. Make two separate C/C++ files for the problems.