National Institute of Technology Calicut
Department of Computer Science and Engineering
Third Semester B. Tech.(CSE)-Monsoon 2024
CS2091E Data Structures Laboratory
Assignment 7

Submission deadline (on or before): 11:59 PM, 22/10/2024

Policies for Submission and Evaluation:

- You must submit all the solutions of this assignment following the below-mentioned guidelines in the Eduserver course page, on or before the submission deadline.
- Ensure that your programs will compile and execute using GCC compiler without errors. The programs should be compiled and executed in the SSL/NSL.
- During the evaluation, failure to execute programs without compilation errors may lead to zero marks for that evaluation.
- Your submission will also be tested for plagiarism, by automated tools. In case your code fails to pass the test, you will be straightaway awarded zero marks for this assignment and considered by the examiner for awarding an F grade in the course. Detection of ANY malpractice related to the lab course can lead to awarding an F grade in the course.

Naming Conventions for Submission

- Submit a single ZIP (.zip) file (do not submit in any other archived formats like .rar, .tar, .gz). The name of this file must be

  `ASSG<NUMBER>_<ROLLNO>_<FIRST-NAME>.zip`

  (Example: *ASSG1_BxxyyyyCS_LAXMAN.zip*). DO NOT add any other files (like temporary files, input files, etc.) except your source code, into the zip archive.
- The source codes must be named as

  `ASSG<NUMBER>_<ROLLNO>_<FIRST-NAME>_<PROGRAM-NUMBER>.c`

  (For example *ASSG1_BxxyyyyCS_LAXMAN_1.c*). If you do not conform to the above naming conventions, your submission might not be recognized by our automated tools and hence will lead to a score of 0 marks for the submission. So, make sure that you follow the naming conventions.

Standard of Conduct

- Violation of academic integrity will be severely penalized. Each student is expected to adhere to high standards of ethical conduct, especially those related to cheating and plagiarism. Any submitted work MUST BE an individual effort. Any academic dishonesty will result in zero marks in the corresponding exam or evaluation and will be reported to the department council for record keeping and for permission to assign an F grade in the course. The department policy on academic integrity can be found at: https://minerva.nitc.ac.in/?q=node/650.

Questions

1. Write a program to compute the Minimum Spanning Tree (MST) of a connected, simple undirected graph G using Prim's algorithm. The program should print the sequence of edges picked by the algorithm and the total edge weight of the spanning tree. Note that the vertices are labeled from 0 to $n - 1$.
   Implement the following functions:

   (a) *Sequence(x)*: Print the sequence in which edges will be picked if x is the starting node.

   (b) *TotalWeight()*: Print the total cost of the minimum spanning tree.

   Input Format

   - The first line contains an integer $n \in [2, 1000]$, denoting the number of vertices in the graph.
   - The subsequent n lines contain an adjacency matrix of size $n \times n$. If two nodes are directly connected by an edge, it is denoted by a positive integer in the matrix; otherwise, it is denoted by 0. The positive integer corresponds to the weight of the edge. Assume that all edge weights will be distinct.
   - Next line consists of a character from the list `['s', 'b', 'e']`.
   - Input `'s'` followed by a positive integer separated by a space calls the function *Sequence(x)*.
   - Input `'b'` calls the function *TotalWeight()*.
   - Input `'e'` terminates the execution of the program.

   Output Format

   - The output (if any) of each command should be printed on a separate line. However, no output is printed for `'e'`.
   - For character `'s'`: Print each edge as a pair of vertices separated by a space followed by the corresponding edge weight in parentheses.
   - For character `'b'`: Print an integer representing the total edge weight of the minimum spanning tree.

   Sample test case 1

   Input:

```
6
0 2 0 3 7 0
2 0 1 0 8 0
0 1 0 4 0 5
3 0 4 0 0 6
7 8 0 0 0 0
0 0 5 6 0 0
s 1
b
e
```

   Output:

```
1 2 (1) 1 0 (2) 0 3 (3) 2 5 (5) 0 4 (7)
18
```

2. Write a program to compute the Minimum Spanning Tree (MST) of a connected, simple undirected graph G using Kruskal's algorithm.

   Input Format

   - The first line contains an integer $n \in [2, 1000]$, denoting the number of vertices in the graph.
   - The subsequent n lines contain the label of the respective node followed by a space separated nodes adjacent to it, sorted in ascending order from left to right, separated by a single space.
   - The next n lines contain the label of the respective node followed by the weights of the edges corresponding to the adjacency list, separated by a single space. The edge weights are real numbers in the range $[-10000, 10000]$. Further, no two edges have the same weight.

   Output Format

   - Print a single line containing the sum of the edge weights of the minimum spanning tree.

   Sample test case 1

   Input:

```
5
0 1 2 3 4
1 0 2
2 0 1 3
3 0 2 4
4 0 3
0 1 7 10 5
1 1 3
2 7 3 4
3 10 4 2
4 5 2
```

   Output:

```
10
```