CSE221
Lab Assignment 06
FALL 2023

Submission Guidelines :

1. You can code all of them either in Python, CPP, or Java. But you should choose a specific language for all tasks.
2. For each task write separate python files like task2.py, task3.py, and so on. For problems that have subproblems, name those like task1A.py, task1B.py, and so on.
3. Add a hand written explanation of 3-4 lines for each of your solutions and submit that as a single document.
4. For each problem, take input from files called **"inputX_Y.txt"** and output at **"outputX_Y.txt"**, where X is the task number and Y is the sample i/o number. For example, for problem 2 sample 1, the input file is this, "input2_1.txt". For problems that have subproblems, name the files like "input1a_1.txt", "output1a_1.txt" and so on. Same for output.
5. For each task include at least one input file (if any) in the submission folder.
6. Finally zip all the files and rename this zip file as per this format:**LabSectionNo_ID_CSE221LabAssignmentNo_FALL2023.zip** [Example:**LabSection01_21101XXX_CSE221LabAssignment06_FALL2023.zip**]
7. Don't copy from your friends.
8. You MUST follow all the guidelines, naming/file/zipping convention stated above.
   Failure to follow instructions will result in a straight 50% mark deduction.

A useful tool for making graphs:

https://csacademy.com/app/graph_editor/

Task 01:[10 Marks]

You are given a weighted, directed graph with N nodes and M edges. Each edge is represented as a triple (u, v, w), where u and v are the nodes connected by the edge and w is the weight of the edge.

Your task is to find the shortest path from a source node S to all other nodes in the graph using **Dijkstra's algorithm.** You should output the shortest distance from the source node to each of the other nodes in the graph. If a node is unreachable from the source node, its distance should be represented as -1.

Input

The first line of the input contains two integers, N and M (1 <= N <= 1000, 1 <= M <= 100000) denoting the number of nodes and edges in the graph, respectively.

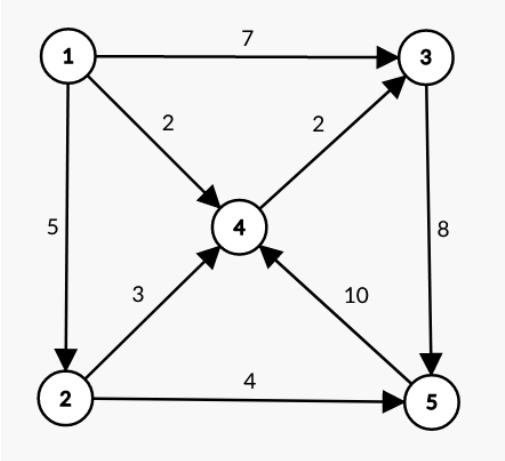
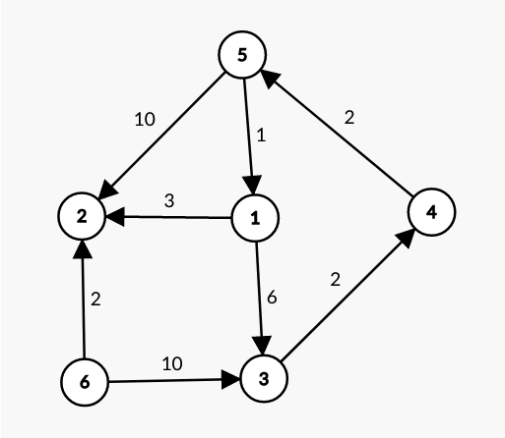
The next M lines each contain three integers, u, v (1 <= u, v <= N), and w (1 <= w <= 100) denoting an edge from node u to node v with weight w.

The last line of the input contains an integer S (1 <= S <= N) denoting the source node.

Output

Output N space-separated integers, where the i-th integer represents the shortest distance from the source node to node i.
If a node is not reachable from the source node, output -1 instead.

Sample Input 1	Sample Output 1	Sample Graph 1

5 8 1 2 5 1 3 7 1 4 2 2 4 3 2 5 4 3 5 8 4 3 2 5 4 10 1	0 5 4 2 9	
Sample Input 2	Sample Output 2	Sample Graph 2
6 8 1 2 3 1 3 6 3 4 2 4 5 2 5 2 10 5 1 1 6 2 2 6 3 10 3	5 8 0 2 4 -1	

Explanation of Sample Input 1

Using Dijkstra's algorithm, the shortest path from node 1 to all other nodes in the given graph is:

- Distance from node 1 to node 1 is 0.
- Distance from node 1 to node 2 is 5.
- Distance from node 1 to node 3 is 4.
- Distance from node 1 to node 4 is 2.
- Distance from node 1 to node 5 is 9.

Task 02:[10 Marks]

Alice and Bob are in a hurry to meet each other and have to traverse through a directed graph with weighted edges. Alice starts from node S and Bob starts from node T. They want to find a common node in the graph, where they can meet each other in the minimum amount of time. Alice or Bob can wait at any node if they want to.

Input

The first line contains two integers N ($1 \leq N \leq 10000$) and M ($1 \leq M \leq 100000$) separated by a space, denoting the number of nodes and edges in the graph, respectively.

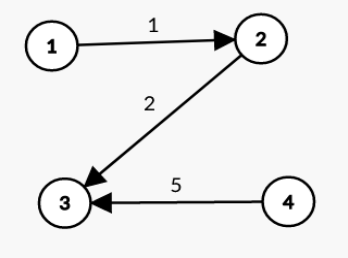
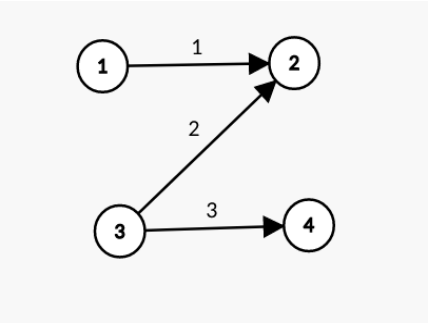
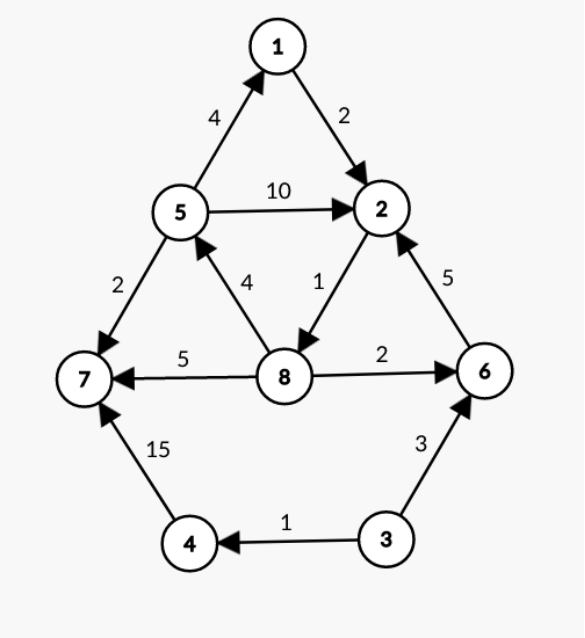
The next M lines each contain three integers, u, v (1 <= u, v <= N), and w (1 <= w <= 100) denoting an edge from node u to node v with weight w.

The last line contains two integers S,T (1 <= S,T <= N) which denotes the starting node of Alice and the starting node of Bob respectively.

Output

Output two integers separated by a space. The first integer is the minimum time it takes for Alice and Bob to meet. The second integer is the vertex where they will meet. If there is no such node, print “Impossible”.

Sample Input 1	Sample Output 1	Sample Graph 1

4 3 1 2 1 2 3 2 4 3 5 1 4	Time 5 Node 3	
Sample Input 2	Sample Output 2	Sample Graph 2
4 3 1 2 1 3 2 2 3 4 3 1 4	Impossible	
Sample Input 3	Sample Output 3	Sample Graph 3
8 12 3 6 3 3 4 1 6 2 5 4 7 15 2 8 1 8 6 2 8 7 5 8 5 4 5 2 10 5 1 4 1 2 2 5 7 2 3 5	Time 8 Node 2	

Task 3 [10 Marks]

Study material: http://www.shafaetsplanet.com/?p=763

There is a group of N people living in a small village. They live in their own house. Although they are all neighbors, they don't all know each other very well.

Each person in that village has their own unique identity - labeled with an integer value between 1 to N. Initially, the villagers don't have any friends. As time passes by, they begin to make friendship between themselves.

In this problem, you will be given a description of K friendships. You have to print an integer value which denotes the size of their friend circle.

Suppose, there are five people living in the village labeled with 1,2,3,4 and 5. Initially, the size of each friend circle is one, since no friendship has been created yet.
One day, person 1 and person 2 become friends. So the size of their friend circle becomes two. Next day, person 3 and person 4 become friends and the size of their friendship becomes two as well. After a few days, person 1 and person 4 become friends. Now the size of their friend circle becomes four consisting of persons 1,2,3 and 4.

Input Format:

The input consists of two integers, separated by a space, denoting the number of people in the village, N ($1 \le N \le 10^5$) and the number of queries that will follow, K ($1 \le K \le 10^5$).

The next K lines contain two integers A_i and B_i each ($1 \le A_i, B_i \le N$ and $A_i \neq B_i$), separated by a space, representing two people who have become friends as a result of the query.

Output Format:

For each query, output a single integer on a new line representing the size of the friend circle that the two people belong to after becoming friends.

Sample Input/Output:

Sample Input 1	Sample Output 1
5 3 1 2 3 4 1 4	2 2 4
Sample Input 2	Sample Output 2
8 7 2 4 4 5 3 6 4 7 3 1 2 7 6 2	2 3 2 4 3 4 7

Sample Input Explanation:

In sample input 2,

Query 0: Initially, there are 8 people in the village who do not know each other.

{1} {2} {3} {4} {5} {6} {7} {8}

Query 1: After person 2 and person 4 becoming friends:

{1} {2,4} {3} {5} {6} {7} {8}

The output is 2, since the size of the friends circle {2,4} is 2.

Query 2: After person 4 and person 5 becoming friends:

{1} {2,4,5} {3} {6} {7} {8}

The output is 3, since the size of the friends circle {2,4,5} is 3.

Query 3: After person 3 and person 6 becoming friends:

{1} {2,4,5} {3,6} {7} {8}

The output is 2, since the size of the friends circle {3,6} is 2.

Query 4: After person 4 and person 7 becoming friends:

{1} {2,4,5,7} {3,6} {8}

The output is 4, since the size of the friends circle {2,4,5,7} is 4.

Query 5: After person 3 and person 1 becoming friends:

{2,4,5,7} {1,3,6} {8}

The output is 3, since the size of the friends circle {1,3,6} is 3.

Query 6: Since the person 2 and person 7 are already in the same friend circle, nothing changes:

{2,4,5,7} {1,3,6} {8}

The output is 4, since the size of the friends circle {2,4,5,7} is 4.

Query 7: After person 6 and person 2 becoming friends:

{2,4,5,7,1,3,6} {8}

The output is 7, since the size of the friends circle {2,4,5,7,1,3,6} is 7.

Task 4 [10 Marks]

In the kingdom of Beluga, there are N cities connected by M roads, each with a maintenance cost associated with it. There is at least one path between any two cities. The king is concerned about the increasing maintenance cost and decides to take action.

He calls upon his council, and they suggest that they find a minimum-cost set of roads that connects all cities while minimizing the maintenance cost. Then the king decides to reduce the total maintenance cost by destroying some of the existing roads, instead of building new ones.

Since you are a very good programmer the king calls you. He asks you to find out what the lowest maintenance cost can be achieved after destroying a few roads while ensuring there still exists a path from each city to another.

Input

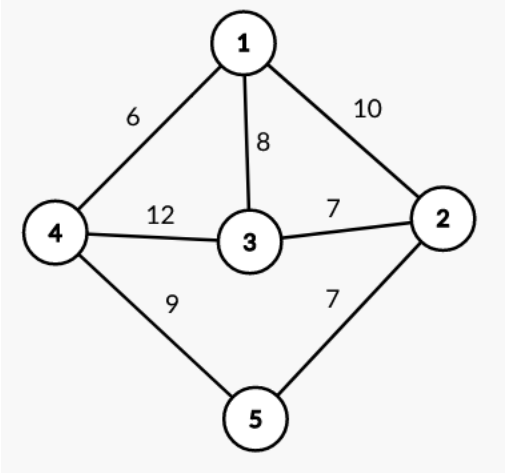
The first line of the input contains two space-separated integers, N and M ($1 \leq N \leq 10^5$, $1 \leq M \leq 10^6$), representing the number of cities and roads in the kingdom of Beluga, respectively.

The next M lines each contain three space-separated integers, u, v, and w ($1 \leq u, v \leq N$, $1 \leq w \leq 10^9$), where u and v denote the endpoints of a road and w represents its maintenance cost.

Output

The output should contain a single integer, the minimum total maintenance cost achievable.

Sample Input/Output:

Sample Input 1	Sample Output 1	Sample Graph 1
5 7 1 2 10 1 3 8 1 4 6 2 3 7 2 5 7 3 4 12 4 5 9	28	
Sample Input 2	Sample Output 2	Sample Graph 2
6 9 1 2 6 2 4 5 2 3 4 1 3 8 4 3 4 2 6 1 2 5 3 5 6 2 5 1 7	17