

Shortest Path

Given a weighted graph (a graph in which each edge has some weight), find the shortest path between two nodes.

(You may use adjacency matrix representation of graph)

Note: The graph is **directed**.

Input:

First Line of the input will contain **Q** representing the number of Queries to be asked on the given graph..

Second line of the Input will contain two space separated integers **V** and **E** representing **Number of Vertices** and **Number of Edges** respectively.

Next **E** lines will contain 3 space separated integers **A**, **B** and **W** representing an edge of weight W between the vertices A and B. E.g., 1 2 3 represents edge between 1 and 2 of weight 3.

Next **Q** lines will contain two space separated integers representing the nodes between which the shortest distance is to be found.

Vertices will be labeled starting from 1. For example if V=5, the set of vertices is {1,2,3,4,5}

Output:

Output a single integer **N** for each test case representing the shortest path length.

Output -1 if it is not possible to go from source node to the destination node.

Constraints:

$$2 \leq V \leq 1000$$

$$0 \leq E \leq V*(V-1)$$

$$1 \leq Q \leq 10$$

$$1 \leq A, B \leq V$$

$$0 \leq W \leq 10000$$

Sample Input

```
2
5 5
1 2 1
1 3 1
3 5 3
1 5 5
5 4 2
1 5
2 3
```

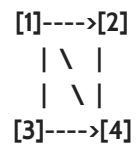
Sample Output:

```
4
-1
```

Explanation

On interpreting the input and constructing the graph we get:

All edges are of length 1 in this case:



Shortest path from 3 to 2 is of length 2: 3→1→2 or 3→4→2

Shortest path from 4 to 1 is of length 1: 4→1