CS 577: Introduction to Algorithms **Program 4 - Vaccination Distribution**

Out: 02/23/21 **Due: 03/02/21**

1 Coding Question:

Reminders

- Must be coded individually in your choice of either Python, Java, C, C++, or C#
- Submitted through Gradescope, and there are hidden test cases
- There is a class-wide runtime leaderboard on Gradescope
- We encourage the use of Piazza for debugging help
- Please do not cheat

Problem

Suppose you are in charge of managing the supply chain system for a vaccine manufacturing company. There is a set n locations, labeled $v_1, v_2, v_3, \ldots, v_n$ that will be used to distribute and give the shots. For each connected pair, v_i and v_j, we have a distance function $d(v_i, v_j)$. For any pair, $d(v_i, v_j) > 0$ and the distances are symmetric, i.e. $d(v_i, v_j) = d(v_j, v_i)$.

We have the capacity to set-up k production centers anywhere that can produce the vaccines, but the vaccine is such that the bottleneck is in transportation between sites rather than the capacity of production. Assign each vaccination distribution location to one of the k production centers so as to minimize the maximum distance between any vaccination distribution locations.

If needed, in a tie-breaking situation, your solution should keep the edge (u, v), $u < v$, over the edge (r, s), $r < s$, when $u < r$ or, if $u = r$, $v < s$.

Input:

- Input should be read in from stdin.
- The first line will contain three space-separated integers n, k, m: the number of vaccination centers, production centers, and edges.
- Each of the following m lines will contain three integers; u, v, and d, where d represents the distance between the nodes u and v. For example, `1 3 55` indicates that the distance between locations 1 and 3 is 55 units. You can assume that the distances are positive, but you should not assume that they are distinct.

Output:

- The output should be written to stdout.
- The output should be the maximum distance in the assignment. If the solution has no edges, output -1.

Constraints:

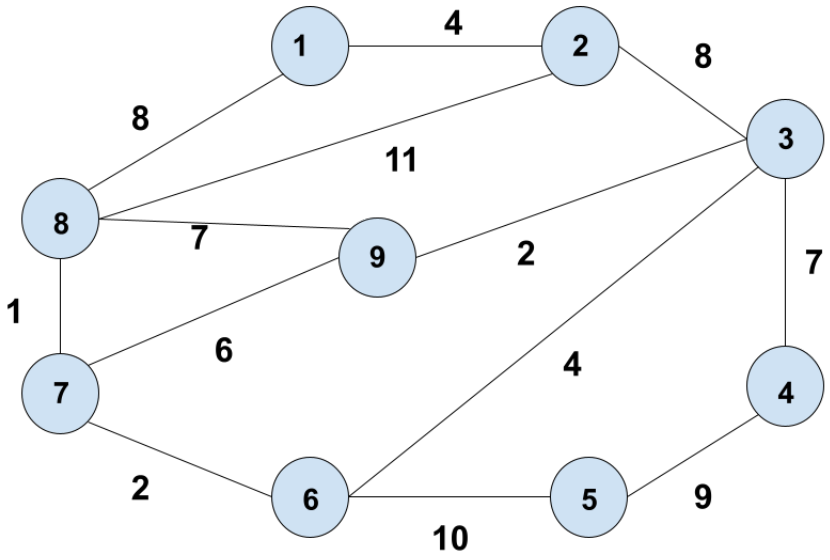
- $2 \leq n \leq 10000$
- $1 \leq k, d \leq 1000$
- $1 \leq m \leq n^2$
- $1 \leq u, v \leq n$

Examples

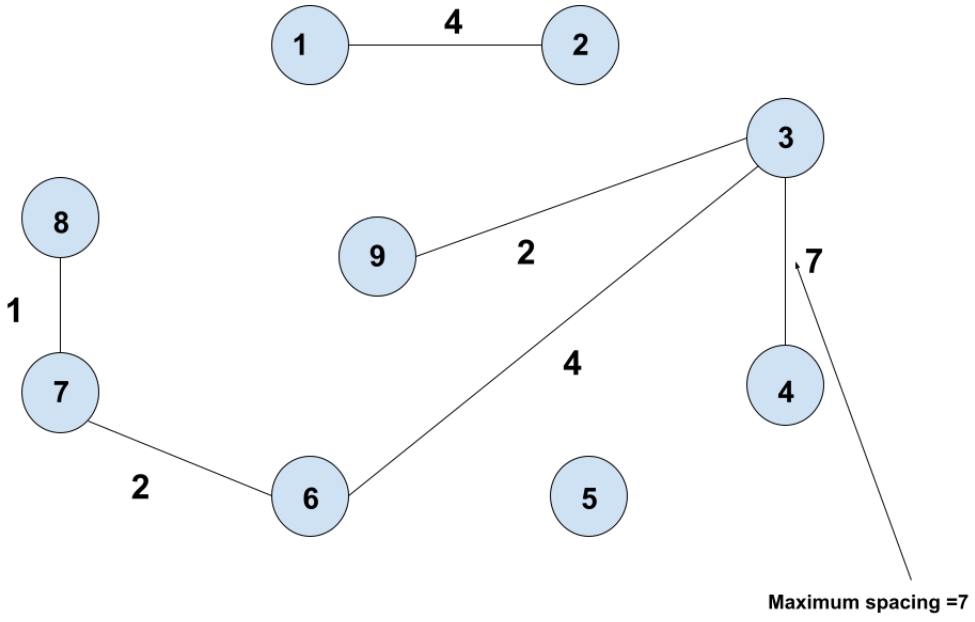
Example 0

```
9 3 13
8 1 8
8 2 11
8 9 7
8 7 1
7 9 6
7 6 2
1 2 4
2 3 8
9 3 2
6 3 4
6 5 10
5 4 9
3 4 7
```

For the following graph:



When $k = 3$, we get:



Example 1
input:

```
5 4 10
1 2 1
1 3 2
1 4 4
1 5 5
2 3 4
2 4 3
2 5 6
3 4 1
3 5 7
4 5 8
```

output:

```
1
```

Example 2
input:

```
6 4 10
1 2 6
1 4 5
1 5 4
2 4 1
2 5 2
2 3 5
2 6 3
3 6 4
4 5 2
5 6 4
```

output:

```
2
```