max planck institut informatik

Competitive Programming

Saarland University — Summer Semester 2020

Julian Baldus, Markus Bläser, Karl Bringmann, Marian Dietz, Simon Schwarz,
Christoph Weidenbach, Dominic Zimmer

Final Exam — Problem Set

July 29, 2020

Please submit solutions to the problems in our judge system, available from within the VM at `https://compro.mpi-inf.mpg.de/`.
You can find your credentials on your personal status page in our CMS.

Problem	buildstair	chase			garlands	park		super	
Points	15	10	10	5	15	10	15	10	10
Difficulty									
Time Limit	1s	2s	2s	2s	1s	5s	5s	1s	1s
Memory Limit	2 GB	2 GB	2 GB	2 GB	2 GB	2 GB	2 GB	2 GB	2 GB

Please note:

- Your solution will be judged immediately after submitting. This may take some time, depending on the current server load.
- You can submit as many times as you want. However, don't abuse the server or try to extract the secret test cases.
- If your solution is **accepted**, you will receive the points specified in the table above.
- If you get **another verdict**, you may be eligible for partial points. Partial points are awarded for the correctness of the idea and the implementation. Please make sure to document your ideas and implementation. Only your last submission in each (sub)task will be considered for partial points.

Space for your wonderful thoughts

Building Stairways (15 points)

Problem ID: buildstair

Your friend Bobby loves to play Minecraft. At least he does on those days when he is allowed to play video games. Today however, he ought to play outside. Defiantly, he grabs his bucket of wooden blocks and heads to play outside.

Bobby is quite experienced in building stairways using all of his blocks. He considers a stairway to be a sequence of blocks of increasing height. As Bobby owns several blocks of equal height, he had to ponder if he wants to leave some blocks out or if he allows the stairway to have level segments. He chose to use *all the blocks* and to allow adjacent blocks of the stairway to have the same height.

Building stairways is quite an easy task, even for Bobby. Therefore, he decided to count the number of possible stairways. You reckon that this will be a very tedious task and decide to help him. How many valid stairway arrangements can you build with Bobby's blocks?

Input

The first line contains an integer n ($1 \leq n \leq 200\,000$). The following line contains n integers h_i ($1 \leq h_i \leq 100$), where h_i is the height of the i-th block.

Output

Print the number of ways Bobby can arrange his blocks into a stairway, modulo $10^9 + 7$.

Explanation of Sample 1

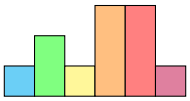

Figure 1: Sample Input 1

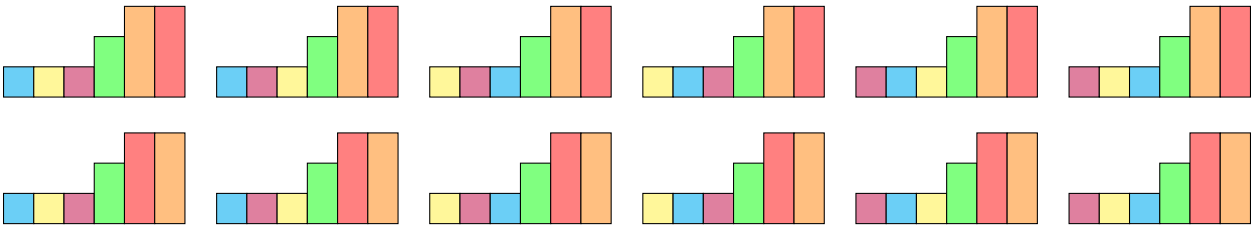

Figure 2: All valid arrangements of Sample Input 1 – 12 in total

Sample Input 1

```
6
1 2 1 3 3 1
```

Sample Output 1

```
12
```

Sample Input 2

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

Sample Output 2

```
1
```

Sample Input 3

```
13
1 1 1 1 1 1 1 1 1 1 1 1 1
```

Sample Output 3

```
227020758
```

Chasing Dieter (25 points)

Problem ID: chase

Dieter Schlau thought running an underground discotheque as his headquarter is a good idea. And indeed, it was – until the police surrounded the place and charged in through the front door.

Dieter has gathered substantial experience in playing cops and robbers, so he was prepared. As soon as an officer entered his office, he knocked him unconscious, stole his uniform and walked straight out of the main entrance – disguised in plain sight. Now, the hunt is on. Dieter arrived at his E-Scooter that conveniently travels just as fast as police cars do. Dieter's goal is obvious: He must arrive at the border before the police does.

Just a few moments later, the police found the unconscious officer, and started the search for Dieter immediately. Knowing Dieter can't be far away, the police shares the news via radio communication: Dieter is on the loose. All k nearby police patrols now try to prevent Dieter from making it to the border. But Dieter – being *Schlau* himself – intercepted the radio communications and now knows where the police patrols are located.

Dieter is wondering: What is the shortest path from Dieter (position 1) to the border (at position V)? Dieter really cannot afford to get caught, so he will only travel along *safe* paths. He considers a path only to be *safe*, if it is guaranteed that he will arrive at each position along the path (strictly) before any police patrol does.

Subtasks

- **Subtask 1** (10 points) $k = 0$, i.e. the police is not actually looking for Dieter. Just find the shortest path to the border.
- **Subtask 2** (10 points) $k = 1$. There is exactly one police patrol looking for Dieter.
- **Subtask 3** (5 points) $0 \leq k \leq V$, i.e. there may be many police patrols.

Input

The first line contains three integers V, E and k ($1 \leq V \leq 100\,000$, $1 \leq E \leq 200\,000$, $0 \leq k \leq V$). The following E lines each contain three integers u, v and l ($1 \leq u, v \leq V$, $1 \leq l \leq 10^9$), describing a one-way road from position u to position v that takes l minutes to use.The last line contains k integers: The initial positions of the police patrols.

It is guaranteed that for any two locations u and v, there is at most one immediate road from u to v. Additionally, all police patrols are located at distinct locations.

Output

If Dieter can find a *safe* path to the border, print the shortest length of such a path. Otherwise, print -1.

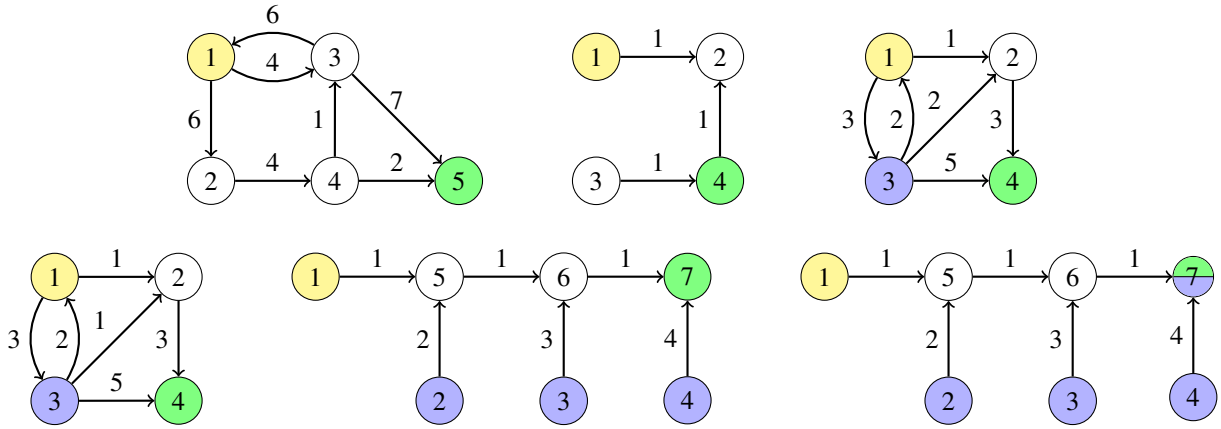

Figure 1: The graphs of Sample Input 1 to 6. Dieter: Yellow, Border: Green, Police: Blue

Subtask 1: Samples 1 and 2 Subtask 2: Samples 3 and 4 Subtask 3: All samples

Sample Input 1

```
5 7 0
1 3 4
1 2 6
2 4 4
4 3 1
3 1 6
4 5 2
3 5 7
```

Sample Output 1

```
11
```

Sample Input 2

```
4 3 0
1 2 1
3 4 1
4 2 1
```

Sample Output 2

```
-1
```

Sample Input 3

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

Sample Output 3

```
4
```

Sample Input 4

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

Sample Output 4

```
-1
```

Sample Input 5

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

Sample Output 5

```
3
```

Sample Input 6

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

Sample Output 6

```
-1
```

Space for your wonderful thoughts

Garlands (15 points)

Problem ID: garlands

You have been invited to two birthday parties that take place at the same time. As a present, you want to bring a garland with the name of the inviter. Currently, you are unsure about which invitation to accept. However, you have to order the garland now, since the delivery takes a long time. You want to leave it open which birthday to attend. Therefore, you want to have the option to gift an appropriate garland for both birthdays.

Now you could simply order two garlands, but this option is too expensive. Therefore, you decide to order a garland that has both names as an *subsequence*. This enables you to rip off some letters to get the appropriate name for the party you are attending. Note that you cannot reorder any letters.

Since you have to pay for each letter, you are interested in the minimum number of letters you have to buy.

Input

The only line of the input contains two strings s_1 and s_2 ($1 \leq |s_1|, |s_2| \leq 1\,000$), the names of the two people who invited you to their birthdays. All characters in the strings are lowercase english characters.

Output

Print the length of the shortest garland that has both s_1 and s_2 as a subsequence.

Explanation of Sample 1

You are invited to the birthday of `dieter` and `petra`. You can order a garland containing `dpietera` and rip off some letters to get a garland for both names.

Sample Input 1	Sample Output 1
`dieter petra`	`8`

Sample Input 2	Sample Output 2
`john tim`	`7`

Sample Input 3	Sample Output 3
`jack jack`	`4`

Sample Input 4	Sample Output 4
`aron nora`	`7`

Amusement Park (25 points)

Problem ID: park

An amusement park is only as good as its worst path. – A Young Philosopher, 2020

An amusement park wants to appeal to new visitors. A recent study interviewed the visitors and assigned each bidirectional path between two attractions a number – the lower the number, the better the path's quality. The study revealed that particularly elder visitors dislike walking along paths of poor quality.

As a consequence of the study, the park management has decided to close some of the worst paths and, if necessary, also some attractions. Up to k attractions can be closed, as well as any number of paths, under the restriction that all remaining attractions must remain reachable from the entrance (which is attraction 1) using open paths only.

Of course, you try your best to close those attractions that require visitors to take paths of poor quality. What is the quality of the worst path that remains?

Subtasks

- **Subtask 1** (10 points) $k = 0$, i.e. all attractions stay opened and must remain reachable.
- **Subtask 2** (15 points) $0 \leq k \leq V - 2$, i.e. it is acceptable that some attractions close and become unreachable.

Input

The first line contains three integers V, E and k ($2 \leq V \leq 100\,000$, $V - 1 \leq E \leq 200\,000$, $0 \leq k \leq V - 2$). The following E lines each contain three integers u, v and q ($1 \leq u, v \leq V$, $1 \leq q \leq 10^9$), describing a bidirectional path between attraction u and attraction v of quality q.

It is guaranteed that (before closing) every attraction is reachable from every other attraction. Additionally, the park neither has paths leading from one attraction to itself nor multiple edges between any pair of attractions.

Output

Print the quality of the worst remaining path. Note that due to $k \leq V - 2$, there is always at least one remaining path.

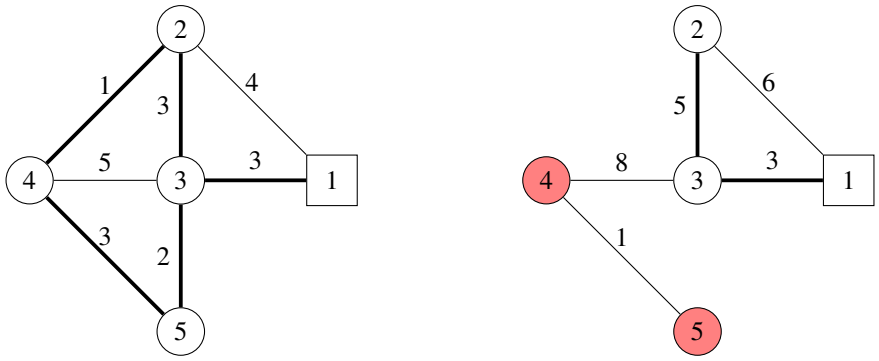

Figure 1: The Graphs of Sample Input 2 and 4. Closed attractions are coloured red, remaining paths are bold

Samples 1 and 2 may occur in both subtasks. Samples 3 and 4 are only valid for the second subtask.

Sample Input 1

```
3 3 0
1 3 5
1 2 3
2 3 4
```

Sample Output 1

```
4
```

Sample Input 2

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

Sample Output 2

```
3
```

Sample Input 3

```
3 3 1
1 2 7
1 3 5
2 3 3
```

Sample Output 3

```
5
```

Sample Input 4

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

Sample Output 4

```
5
```

Space for your wonderful thoughts

Super Mario (20 points)

Problem ID: super

You are playing *New New New Super Mario Bros* and are trying to beat the highscore in an unconventional level.

The level consists of n platforms numbered from 1 to n, floating at different heights h_i ($1 \leq i \leq n$). At the beginning of the level, you are equipped with a Tanooki Suit that allows you to jump up to d units high. Due to its aerodynamic properties, the suit allows you to jump over arbitrarily wide horizontal gaps. Additionally, the suit allows you to land safely from any drop of height at most d. You cannot jump higher than d units, and if you fall from more than d units, you die.

Furthermore, being a game of the Mario franchise, you can only move from left to right. That is, you can only jump from pillar i to pillar j if $j > i$.

You may start the level on any platform of your choice. What is the longest sequence of jumps you can achieve?

Subtasks

- **Subtask 1** (10 points) $1 \leq h_i \leq 100$, $1 \leq d \leq 100$.
- **Subtask 2** (10 points) $1 \leq h_i \leq 100\,000$, $1 \leq d \leq 100\,000$

Input

The first line contains two integers n and d ($1 \leq n \leq 100\,000$). The following line contains n integers: the vertical positions of the individual platforms h_i.

Output

Print the length of the longest possible jump sequence.

Figure 1: Possible Solutions of Sample Input 1

Samples 1 and 2 are valid for both subtasks. Sample 3 is only valid for subtask 2.

Sample Input 1

```
5 2
1 6 3 5 2
```

Sample Output 1

```
3
```

Sample Input 2

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

Sample Output 2

```
5
```

Sample Input 3

```
9 1
101 102 103 115 114 113 112 111 104
```

Sample Output 3

```
5
```

Space for your wonderful thoughts