max planck institut informatik

Competitive Programming

Saarland University — Summer Semester 2022

Markus Bläser, Karl Bringmann, Martin Bromberger, Christoph Weidenbach
Julian Baldus, Yasmine Briefs, Simon Schwarz, Dominic Zimmer

Assignments Week 11

Deadline: **July 21, 2022 at 16:00 sharp**

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

Problem	`minhampath`	`influencer`	`treematching`	`cocktail`
Points	3	3	3	3
Difficulty				

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 will receive 0 points.

Hamiltons Race

Problem ID: minhampath
Time limit: 3 seconds

Currently, the Formula One is taking place, and the next Grand Prix race is almost fully planned. The location is already settled. The racetrack there consists of n edges and m one-way road segments between them. However, the exact route is not yet fixed. The organisers want the track to start and end at any node and pass through *all nodes* exactly once, such that all fans can have a look at their idols.

Lewis Hamilton wants to win this Grand Prix. Fortunately, Lewis knows the committee in charge of selecting the next race track, and can ask them to select an appropriate route. Lewis, still a little drunk from the party after his last win, has the following plan: "If the route is shortest, I will reach the goal fast and, thus, win!".

Can you help Lewis by finding the shortest possible route that visits each vertex once? The route may start and end at any node (which then count as visited). The length of the route is defined as the sum of the lengths of all segments.

Input

The first line of each test case consists of the two integers n and m ($2 \leq n \leq 20, 1 \leq m \leq n^2$). Then, m lines with three integers u, v, w representing an edge from u to v with length w ($1 \leq w \leq 10^9$). Note that there may be self-loops or multiple edges between u and v.

Output

Print a single integer: The length of the shortest valid path. If no path that visits all nodes exists, output `impossible`.

Sample Input 1

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

Sample Output 1

```
16
```

Sample Input 2

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

Sample Output 2

```
impossible
```

Sample Input 3

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

Sample Output 3

```
2
```

Influencer

Problem ID: influencer
Time limit: 3 seconds

You want to market your new Waldhorn. To kickstart your Horn, you hire k influencers to play and promote your instrument.

Group dynamics on Instrugram work as follows: Each person has a group of influencers they follow. A person will only buy – and then play – your intrument, if all of the influencers they are following are playing the instrument, too. Otherwise, they will not buy your instrument.

First, all hired influencers will receive one Waldhorn. Next, all people that are now convinced (since all of their influencers are playing it) will buy your Waldhorn. Those people may be influencers for other people as well. This process continues iteratively. How many people will eventually play the Waldhorn?

Input

The first line contains two integers n and m ($1 \leq n, m \leq 10^6$), the number of people and the number of Instrugram-relationships.

The following m lines each contain two integers a and b ($a \neq b$), describing that person a follows person b on Instrugram.

It is guaranteed that every person follows at least one other person.

The last line contains an integer k. Next k integers x_i follow, indicating that you hired the x_i-th person as an influencer.

Output

Print the number of people who will play the Waldhorn.

Sample Inputs

Sample Input 1

```
6 9
1 2
1 5
2 3
2 4
2 6
3 4
5 1
4 1
6 1
2 4 6
```

Sample Output 1

```
4
```

Sample Input 2

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

Sample Output 2

```
1
```

Sample Input 3	Sample Output 3
3 6 1 2 1 3 2 3 2 1 3 1 3 2 2 1 2	3

Exact Tree Matching

Problem ID: treematching
Time limit: 2 seconds

You are given a rooted tree and a pattern. Your task is to determine if the pattern matches the given tree. A tree T matches a pattern if and only if there is a vertex $v \in T$ such that the subtree starting at v is identical to the pattern. In particular, the subtree and the pattern must have the exact same structure and same labels.

The given tree and pattern is *ordered*. If a node has multiple children, they are ordered by their respective ID (not label) in increasing order. Two trees are only equal if the ordering in both trees is equal.

Input

The input contains a tree and a pattern. Both are specified in the following way:

The first line of the description contains an integer n ($2 \leq n \leq 10^6$), the number of nodes in the tree/pattern. Nodes in the tree have numbers 1 to n, where the i-th node of the tree has ID i. The next line contains n integers l_i ($1 \leq l_i \leq 10^6$), the label of the i-th node in this tree/pattern. In the third line, there are $n-1$ integers p_i ($1 \leq p_i \leq n$), where p_i denotes the ID of the parent of node $i+1$. Note that node 1 has no parent, as it is the root of the tree/pattern.

Overall, lines 1–3 of the input correspond to the tree, lines 4–6 correspond to the pattern.

Output

Print `yes` if the pattern matches the given tree, `no` otherwise.

Sample Inputs

Note that in sample input 4, the pattern does not match because of the different ordering.

Sample Input 1

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

Sample Output 1

```
yes
```

Sample Input 2

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

Sample Output 2

```
no
```

Sample Input 3

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

Sample Output 3

```
no
```

Sample Input 4

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

Sample Output 4

```
no
```

Sample Input 5

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

Sample Output 5

```
yes
```

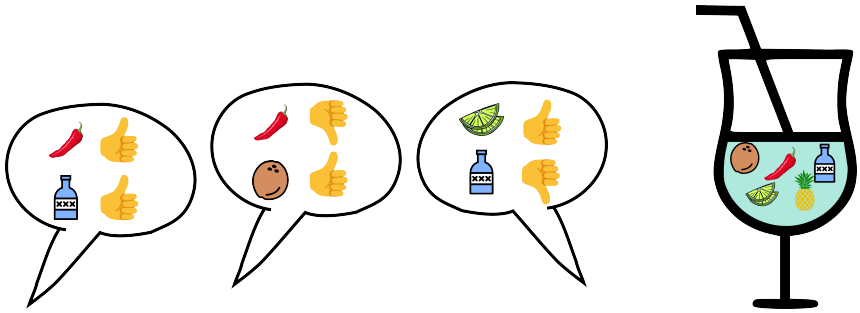
Cocktail

Problem ID: cocktail
Time limit: 1 second

Dieter's new discotheque was a huge success. Particularly popular is the cocktail bar. There, of course, you can order the classics milk, tea and tea-killer-milk-rice. While that's enough to satisfy the regular crowd, Dieter has to come up with something new for the city's gangsters, who still prefer to hang out elsewhere.

So Dieter decides to offer a brand new cocktail. He already knows the gangsters' preferences from his underground spies. Each of them has two preferences concerning the ingredients of cocktails. Preferences can express both likes and dislikes. Examples of preferences would be "I love cocktails with coconut" (i.e. a like of the ingredient coconut) or "Chilli does not belong in a cocktail" (clearly a dislike of the ingredient chilli).

As Dieter only wants to add one cocktail to his bar, it is hard to satisfy all preferences. Therefore, Dieter would like his new cocktail to satisfy at least one preference from each gangster. He hopes that will be enough to convince them all to come to his disco.



Depiction of Sample Input 1

Input

The first line contains two integers n and m ($1 \leq n \leq 10^5$, $1 \leq m \leq 2 \cdot 10^5$), the number of gangsters and the number of cocktail ingredients. The cocktail ingredients are numbered $1, \ldots, m$.

The following n lines each describe the two preferences of one gangster. Each of these lines is of the form $c_1\ i_1\ c_2\ i_2$ ($c_1, c_2 \in \{'+','-'\}$, $1 \leq i_1, i_2 \leq m$). A '$+$' means that the gangster likes the respective ingredient, while a '$-$' means that the gangster dislikes the respective ingredient.

For example, $+\ 5\ -\ 2$ means that the gangster would like to have ingredient 5, but would prefer not to have ingredient 2 in the cocktail.

Output

If it is impossible to mix a valid cocktail (ie. a cocktail that satisfies at least one preference of each of the n gangsters), ouput `IMPOSSIBLE`.

Otherwise, output the description of a valid cocktail: Print a line consisting of m symbols, the i-th of them being '$+$' if the i-th ingredient should be included in the cocktail, and '$-$' if the i-th ingredient should not be included in the cocktail.

If there are multiple valid cocktails, output any of them.

Sample Inputs

In the first sample, a valid cocktail could consist of the ingredients $2, 3$ and 4. The first gangster is happy because ingredient 2 is in the cocktail. For the second gangster, we even satisfied both preferences, and the third gangster is happy because ingredient 4 is in the cocktail.

For sample input two, no valid cocktail exists. In order to safisfy the first gangster, we definitely have to include ingredient 1, but if we want to satisfy the second gangster, we cannot include ingredient 1.

In the third sample, a cocktail with no ingredients will satisfy all gangsters.

Sample Input 1

```
3 5
+ 1 + 2
- 1 + 3
+ 4 - 2
```

Sample Output 1

```
+ + + + +
```

Sample Input 2

```
2 1
+ 1 + 1
- 1 - 1
```

Sample Output 2

```
IMPOSSIBLE
```

Sample Input 3

```
3 2
+ 1 - 2
- 1 + 2
- 1 - 2
```

Sample Output 3

```
- -
```