CS3230 Design and Analysis of Algorithms

Programming Assignment 2

Released October 12, 2020 8:00 AM

Due November 13, 2020 11:59 PM

Problems

Uprooted (60 marks)

Time limit: 2 seconds
Memory limit: 512 MB

The Tree of Harmony, a magical tree that protected Equestria from its enemies, was destroyed in the recent war with King Sombra. Without it, Equestria is the most vulnerable it has ever been.

Though the Tree of Harmony may be physically gone, its spirit lives on in a group of tight-knit friends.

Six friends—Gallus, Ocellus, Sandbar, Silverstream, Smolder and Yona—were called to commemorate the Tree of Harmony. To keep its memory alive, they will work together to build a model of what was Equestria's most powerful protector.

They unfortunately do not recall exactly how the Tree of Harmony looked like, but they have shared their ideas with each other. After much deep thought, they have reached a consensus on, and are *sure* of, the following:

1. It was possible to model the Tree of Harmony as n junctions, labeled from 1 to n, connected by branches, such that it was possible to trace a path from any junction to any other junction by using the branches and cutting any single branch made this no longer true.
2. The Tree of Harmony was rooted on the ground at the junction labeled 1.
3. The Tree of Harmony satisfied s known *growth sequences*.
    - The i^{th} of these growth sequences is a permutation p_i of the numbers from 2 to n.
    - Consider the i^{th} growth sequence p_i, and consider the following process:
        (a) Maintain a set w of junctions, originally containing just the root.

(b) Process the elements of p_i in the order $p_{i,1}, p_{i,2}, \ldots, p_{i,n-1}$.

(c) When processing the element $p_{i,j}$, there must be at least one branch connecting the junction labeled $p_{i,j}$ and some junction in w. Add the junction labeled $p_{i,j}$ to w.

Satisfying p_i means that the condition in the final step is always true—at least one required branch always exists.

They have not reached an agreement of which branches were actually on the Tree of Harmony, but they have a list of m candidates. The i^{th} of these candidate branches connects junctions a_i and b_i and has a *level of confidence* c_i. This value represents the group's level of confidence that this branch really existed on the Tree of Harmony, such that a larger level of confidence represents more confidence in this branch's existence.

Now, they wish to construct a model of the Tree of Harmony, consistent with the information they have, by selecting a subset of the candidate branches such that the sum total of the levels of confidence of the selected branches is maximized.

Help them accomplish this!

Input

The first line of input contains three integers, n, m and s, the number of junctions, the number of candidate branches and the number of growth sequences, respectively.

The next m lines of input contain the descriptions of the candidate branches. In particular, the i^{th} of these lines contains three integers a_i, b_i and c_i, denoting that the i^{th} candidate branch connects junctions a_i and b_i and has a level of confidence c_i.

The next s lines of input contain the descriptions of the growth sequences. In particular, the i^{th} of these lines contains $n-1$ integers $p_{i,1}, p_{i,2}, \ldots, p_{i,n-1}$, describing the i^{th} growth sequence.

Output

If there is no way to select a subset of the candidate branches to form a model of the Tree of Harmony consistent with the information they have, output a single line containing the string `IMPOSSIBLE`.

Otherwise, on the first line, output a single integer r, the number of branches they should select.

On the second line, output r integers $l_1, l_2, \ldots, l_r$ ($1 \leq l_i \leq m$), denoting the branches that should be selected. All l_i should be distinct and selecting these branches should result in a model of the Tree of Harmony that is consistent with the information they have, and, among all such consistent models, has the maximum possible sum total of levels of confidence. You can output the branches in any order.

If there are multiple correct answers, you can output any of them.

Scoring

For all groups,

- $2 \leq n$;
- $1 \leq m, s$;
- $(n-1) \cdot s \leq 200\,000$;
- $1 \leq a_i, b_i \leq n$;
- $a_i \neq b_i$;
- $a_i \neq a_j$ or $b_i \neq b_j$ for any $i \neq j$;
- $a_i \neq b_j$ or $b_i \neq a_j$ for any $i \neq j$;
- $1 \leq c_i \leq 10^9$;
- $2 \leq p_{i,j} \leq n$;
- $p_{i,j} \neq p_{i,k}$ for any $j \neq k$;
- $p_i \neq p_j$ for any $i \neq j$.

Group	Marks	n	m	s	Additional constraints
1	6	$n \leq 15$	$m \leq 20$	$s \leq 3$	The inputs for this group are public.
2	10	$n \leq 60$	$m \leq 60$	$s \leq 60$	
3	10	$n \leq 200\,000$	$m \leq 200\,000$	$s = 1$	
4	10	$n \leq 200\,000$	$m \leq 200\,000$	$s \leq 2$	
5	24	$n \leq 200\,000$	$m \leq 200\,000$	$s \leq 200\,000$	

Example

Input 1

```
9 11 2
1 2 12
2 3 7
2 6 10
2 7 3
3 4 6
3 5 4
4 5 5
4 9 7
6 7 4
6 8 5
6 9 4
2 6 9 3 5 4 8 7
2 3 4 5 6 7 8 9
```

Output 1

```
8
1 2 3 5 6 9 10 11
```

Input 2

```
3 1 1
1 2 10
2 3
```

Output 2

```
IMPOSSIBLE
```

Input 3

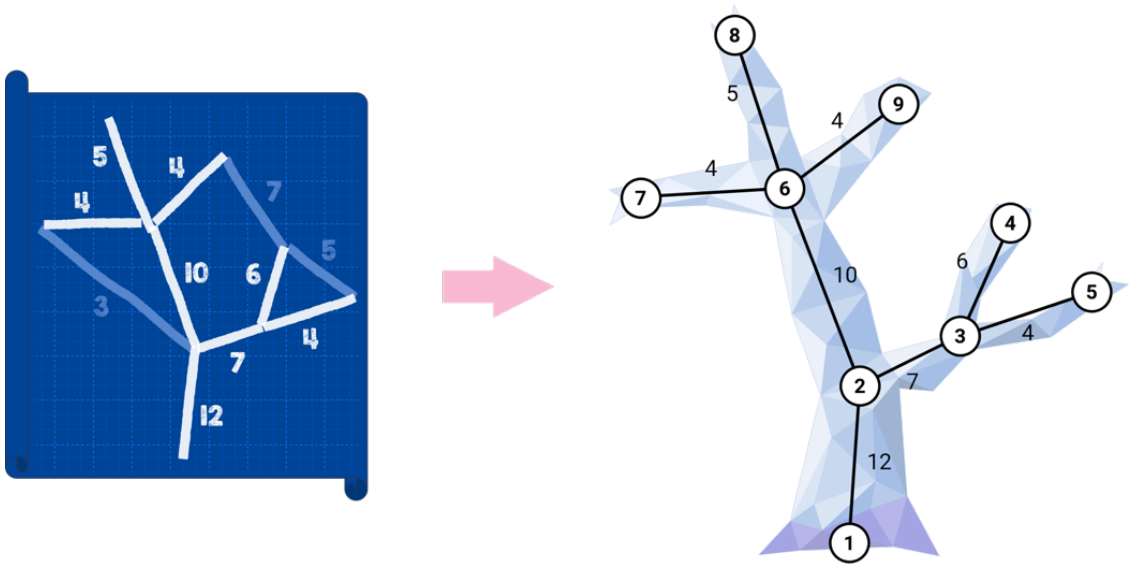
```
5 6 1
1 2 5
1 3 6
2 3 7
2 4 8
3 4 9
4 5 10
2 3 5 4
```

Output 3

```
IMPOSSIBLE
```

Note

The following image illustrates the first example.



Notice that this satisfies both growth sequences.

It satisfies the first growth sequence:

1. We maintain the set w, which initially contains just the junction 1; i.e. $w = \{1\}$.
2. $p_{1,1} = 2$, so there must be some branch connecting junction 2 with some junction in w. We have the branch connecting junctions $1 \in w$ and 2, so this is satisfied. Now $w = \{1, 2\}$.
3. $p_{1,2} = 6$, so there must be some branch connecting junction 6 with some junction in w. We have the branch connecting junctions $2 \in w$ and 6, so this is satisfied. Now $w = \{1, 2, 6\}$.
4. $p_{1,3} = 9$, so there must be some branch connecting junction 9 with some junction in w. We have the branch connecting junctions $6 \in w$ and 9, so this is satisfied. Now $w = \{1, 2, 6, 9\}$.
5. $p_{1,4} = 3$, so there must be some branch connecting junction 3 with some junction in w. We have the branch connecting junctions $1 \in w$ and 3, so this is satisfied. Now $w = \{1, 2, 3, 6, 9\}$.
6. $p_{1,5} = 5$, so there must be some branch connecting junction 5 with some junction in w. We have the branch connecting junctions $3 \in w$ and 5, so this is satisfied. Now $w = \{1, 2, 3, 5, 6, 9\}$.
7. $p_{1,6} = 4$, so there must be some branch connecting junction 4 with some junction in w. We have the branch connecting junctions $3 \in w$ and 4, so this is satisfied. Now $w = \{1, 2, 3, 4, 5, 6, 9\}$.

8. $p_{1,7} = 8$, so there must be some branch connecting junction 8 with some junction in w. We have the branch connecting junctions $6 \in w$ and 8, so this is satisfied. Now $w = \{1, 2, 3, 4, 5, 6, 8, 9\}$.
9. $p_{1,8} = 7$, so there must be some branch connecting junction 7 with some junction in w. We have the branch connecting junctions $6 \in w$ and 7, so this is satisfied. Now $w = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

At every point, a required branch was found, so we have shown that this model of the Tree of Harmony satisfies the first growth sequence. A very similar analysis can be done to show that it satisfies the second growth sequence.

The sum total of the levels of confidence in the branches of this tree is 52. It can be shown that no model of the Tree of Harmony consistent with the information they have has a strictly larger sum total, so this is a correct answer.

For the second example, it is obvious that no subset of the candidate branches will form a tree, so the answer is clearly `IMPOSSIBLE`.

For the third example, it is obvious that no subset of the candidate branches will form a model of the Tree of Harmony satisfying p_1, so the answer is clearly `IMPOSSIBLE`.

The four inputs for **Group 1** have been made public to assist you with debugging. They are accessible here:

https://drive.google.com/drive/folders/12EQl8XSSdY6KBz5CUuI7kekiZeZDXKkf

Each input is a `.in` file, which is just a text file that can be opened by most text editors. The corresponding outputs are provided for the first three tests (which are the examples provided here) but is not provided for the fourth test.

It is possible to obtain the nominal 6 points by solving these inputs manually and hardcoding the answers into your program, but we suggest using them to construct and debug a more general solution. You are free to discuss the expected outputs for these inputs with your coursemates or teaching assistant if you have uncertainties about your understanding of the problem.

Frenemies (40 marks)

Time limit: 2 seconds
Memory limit: 512 MB

Grogar was a legendary tyrant who once wreaked havoc across Equestria. Ages ago, his mythical bell and source of power was stolen and hidden at the top of the treacherous Mt. Everhoof, and he was forced to go into hiding.

At long last, he has decided that the time is ripe for revenge. Even in his weakened form, he coerced three hardened criminals to team up and retrieve his bell; with it, all of Equestria will kneel before him once again.

Cozy Glow, Lord Tirek and Queen Chrysalis, three villains with nothing in common other than their lust for power, now face their toughest challenge yet: working together.

Through their individual efforts, they have each managed to reach the highest level of the mountain. Recovering the bell, however, will require them to combine their powers.

The highest level of Mt. Everhoof can be represented as a grid with r rows and c columns. Each of the cells in this grid is either:

- solid ground, which can freely be entered;
- a huge rock, which is an impassable obstacle that cannot be entered;
- a hole, which when entered will result in certain death;
- a hostile guard, which when entered by Lord Tirek or Queen Chrysalis will result in certain death, but when entered by Cozy Glow is subdued and permanently becomes solid ground;
- a lethal magic barrier, which when entered by Cozy Glow or Queen Chrysalis will result in certain death, but when entered by Lord Tirek is subdued and permanently becomes solid ground;
- an Ophiotaurus, which when entered by Cozy Glow or Lord Tirek will result in certain death, but when entered by Queen Chrysalis is subdued and permanently becomes

solid ground;

- slippery ice, which when entered from a certain direction will result in sliding to the next cell in that direction, unless the next cell is a huge rock, in which case one stops on this cell. Note that it is possible to slide into a lethal obstacle or fall off the mountain (if the next cell is out of the grid), and both cases will result in certain death;
- the cell containing Grogar's bell. In order to successfully retrieve the bell, all three of the villains need to be on this cell simultaneously. For all other purposes, it can be treated as solid ground.

In one move, you can direct a character to move in one of the four cardinal directions. The three characters initially begin in different cells, but they are allowed to occupy the same cell at the same time and will not obstruct each other's movements.

As you can tell, they are new to this whole "cooperation" thing, so they will need a bit of help. Can you give them a list of moves that will allow them to successfully retrieve Grogar's bell?

If the task is truly impossible, you must also say so.

Input

The first line of input contains two integers, r and c, the number of rows and the number of columns of the grid, respectively.

The next $r + 2$ lines each contain $c + 2$ characters. In particular, the j^{th} character on the i^{th} of these lines is either:

- `+`, if either $i = 1$ or $i = r + 2$, and at the same time either $j = 1$ or $j = c + 2$;
- `-`, if either $i = 1$ or $i = r + 2$, and at the same time $j \neq 1$ and $j \neq c + 2$;
- `|`, if either $j = 1$ or $j = c + 2$, and at the same time $i \neq 1$ and $i \neq r + 2$;

  Note that the above three characters are purely decorative. They serve only to clarify the boundaries of the grid. In particular, they are not walls, and they are not cells.
- if none of the above are satisfied, then this character represents the content of the cell in row $i - 1$ and column $j - 1$, and is either:
  - a space, which represents solid ground;
  - `c`, which represents Cozy Glow's initial position and is solid ground;
  - `l`, which represents Lord Tirek's initial position and is solid ground;
  - `q`, which represents Queen Chrysalis's initial position and is solid ground;
  - `#`, which represents a huge rock;
  - `0`, which represents a hole;

- `C`, which represents a hostile guard;
- `L`, which represents a lethal magic barrier;
- `Q`, which represents an Ophiotaurus;
- `.`, which represents slippery ice;
- `%`, which represents the cell containing Grogar's bell.

Output

If it is not possible to retrieve the bell, output a single line containing the string `IMPOSSIBLE`.

Otherwise, on the first line, output a single integer m ($1 \leq m \leq 2 \cdot 10^6$), the number of moves in your solution.

Then, on the next m lines, output the moves in your solution. In particular, the i^{th} of these lines should contain one of `c`, `l` or `q` followed by one of `^`, `<`, `v` or `>`, separated by a space, denoting that the i^{th} move should involve Cozy Glow, Lord Tirek or Queen Chrysalis moving up, left, down or right, respectively.

Note that you *do not need* to minimize m.

It is legal to attempt to move into a huge rock, although nothing will happen. It is also legal to perform more moves even after all three characters have reached the cell containing Grogar's bell, as long as all three characters are at the cell containing Grogar's bell at the end of all the moves.

If there are multiple correct answers, you can output any of them.

Scoring

For all groups,

- $1 \leq r, c$;
- $4 \leq rc \leq 300\,000$;
- It is guaranteed that there is exactly one each of `c`, `l`, `q` and `%` in the grid;
- The input will be such that if a solution exists, then a solution exists with $m \leq 2 \cdot 10^6$.

Group	Marks	Additional constraints
1	13	There will be no `C`, `L` or `Q` in the grid.
2	12	There will be no `L` or `Q` in the grid.
3	15	

Examples

Input 1

```
4 9
+---------+
|cO...  l |
|   # .###|
|##LC .C q|
|%Q   #   |
+---------+
```

Output 1

```
32
l <
l <
l v
l <
l v
l <
l <
l ^
c v
c >
c >
c v
c >
c >
c >
q <
q <
q <
q v
q <
q <
q <
q <
l v
l <
l <
c <
c v
c <
c <
```

```
c <
c <
```

Input 2

```
2 4
+----+
|lcq |
|###%|
+----+
```

Output 2

```
9
l >
l >
l >
l v
c >
c >
c v
q >
q v
```

Input 3

```
2 4
+----+
|lcq.|
|###%|
+----+
```

Output 3

```
IMPOSSIBLE
```