Belgium Algorithm Contest

Belgium Algorithm Contest
Round 2 - 2017

Do not open before the start of the contest.

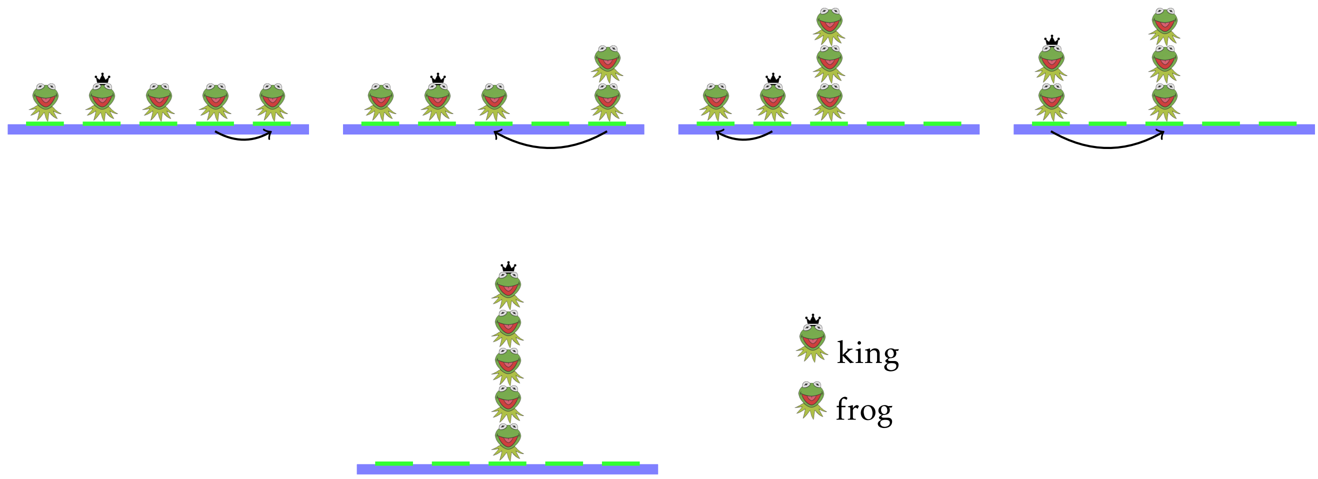
This page is not really blank.

Problem A
Frogs
Time limit: 2s

The frog king wants to meet with all the frogs in his kingdom. The frog kingdom consists of n water lilies aligned on top of a lake and there is one frog living on each of those water lilies (including the king).

To meet each other, the frogs can jump from one water lily to the other until they are all on top of the same one. A frog alone can only jump to a **non-empty** water lily that is immediately next to him. Two frogs can jump to any **non-empty** water lily that is **exactly** 2 water lilies away. In general, if k frogs are on the same water lily, they can jump to any **non-empty** water lily that is **exactly** k water lilies away (the number of water lilies between the origin and the destination is $k-1$).

Also, the king must always remain on top. It is the king after all. The following picture shows an example with $n = 5$ of how the frogs can meet.



The king plans to expand his kingdom and therefore would like your help to figure out how they can meet for any number n and position of the king k.

Input

The input contains a single line with two integers n and k. The first one represents the number of water lilies in the kingdom and k represents the position of the king. If $k = 1$ then the king is on the first water lily and so on.

Constraints

- $1 \le n \le 20000$
- $1 \le k \le n$

Output

If it is possible for all the frogs to meet, print $n - 1$ lines giving the jumps that should be performed with the following format:

```
move i to j
```

where i is the origin water lily and j is the destination water lily. If it is impossible for all the frogs to meet, print a single line with the word `impossible`.

If there are several possible solutions, any one will be accepted.

Input 1

```
5 2
```

Output 1

```
move 4 to 5
move 5 to 3
move 2 to 1
move 1 to 3
```

Input 2

```
6 3
```

Output 2

```
move 3 to 4
move 4 to 2
move 2 to 5
move 5 to 1
move 1 to 6
```

Belgium Algorithm Contest

Problem B
Movies Night
Time limit: 3s

Bob, Alice and Craig want to go to the movies this weekend. Because they have different tastes in movies, Bob has devised the following algorithm to select the movie:

1. Each person sorts the n movies from the one they want the most to the one they want the least.
2. Loop over the lists and give $n-i$ points to the i-th movie (i starting at 0).
3. Sort the movies by points and select the first one.

Whenever the movies are sorted, ties are broken by alphabetical order of the movies names.

For example, assume there are 5 movies A, B, C, D and E with the following rankings:

Score	5	4	3	2	1
Bob	B	C	E	D	A
Alice	E	A	B	D	C
Craig	D	C	A	B	E

Adding the scores will give the following totals:

	Bob	Alice	Craig	Total
A	1	4	3	8
B	5	3	2	10
C	4	1	4	9
D	2	2	5	9
E	3	5	1	9

Therefore, the movie that is selected is B. Craig is not very happy with this choice because B is his second to last choice. He wonders if he could eventually lie about his preferences so that the movie that is selected is as high as possible in his list.

For instance, suppose that instead he give the following fake ranking of his movies.

Score	5	4	3	2	1
Craig	C	A	D	E	B

With this new order the finals scores become:

	Bob	Alice	Craig	Total
A	1	4	4	9
B	5	3	1	9
C	4	1	5	10
D	2	2	3	7
E	3	5	2	10

With these scores both movies C and E have a maximum score of 10. But since C is lexicographically smaller, it will be picked over E. In this case he got his second favorite choice with respect to *his real ranking*.

He was wondering, given the rankings of Bob and Alice and knowing his own real ranking, which fake ranking should he give in order the get the best possible movie with respect to his real ranking?

Note that he does not need to lie, he is allowed to give his true order if that order yields the best movie for him.

Input

The first line of the input contains a single integer n, giving the number of movies.

The next line contains n strings separated by single spaces giving the names of the moves.

The last three lines each contain n strings giving the ordering of the movies for Bob, Alice and Craig respectively.

Constraints

- $3 \leq n \leq 10$
- No two movies have the same name
- Movie names only contain characters from a to z.

Output

A single line with the best movie that Craig can get, with respect to his original ranking, if he gives some ordering of his movies that is not necessarily his real preferred order.

Input 3

```
5
attica ballerina cash dale earth
ballerina cash earth dale attica
earth attica ballerina dale cash
dale cash attica ballerina earth
```

Output 3

```
cash
```

Input 4

```
6
attica ballerina cash dale earth fake
ballerina cash earth fake attica dale
cash dale ballerina attica fake earth
earth dale fake ballerina cash attica
```

Output 4

```
ballerina
```

This page is not really blank.

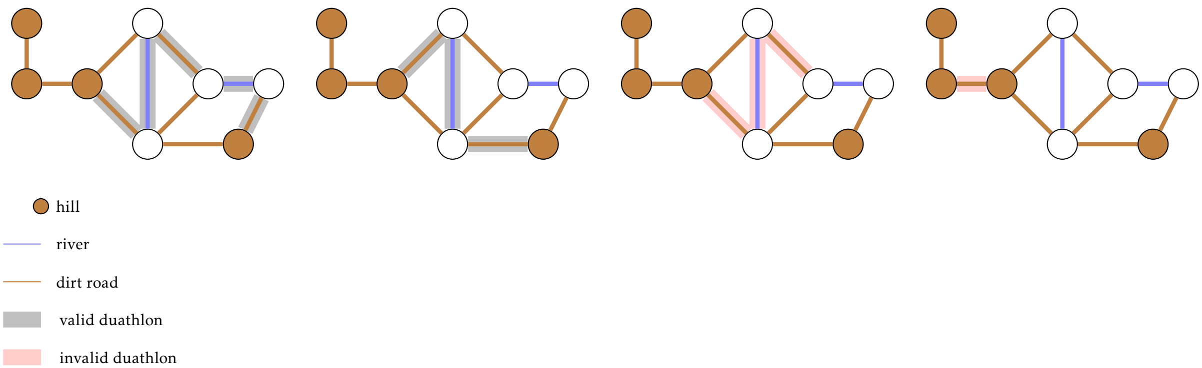
Belgium Algorithm Contest

● Problem C
Duathlon
Time limit: 2s

Bob is in charge of planning the school's duathlon, a race where the participants alternate between running and swimming. He was given a map of where the duathlon will take place and his job is to plan the path that the participants will have to follow.

The map consists of several locations that are connected by either dirt paths or rivers. Now he has to find a path connecting two hills that starts with a dirt road, then goes along a river, then dirt again, and so on. A hill is defined as a location without a river coming out of it. After a careful observation of the map, Bob noticed that there is no location that is connected to more than one river.

The following figure shows a few examples of valid and invalid duathlons. Note that a duathlon must contain at least one dirt section and one river section.



He is having some trouble to find if such a path even exists. Could you help him?

Note: A path cannot visit the same location more than once. This also means that it must start and end at two distinct hills.

Input

The first line of the input contains two integers n and m giving the number of locations and connections between the roads, respectively.

Then follow m lines each with two integers x, y and a character c separated by single spaces. Each such line represents that locations x and y are connected by a either a dirt road if $c =$ 'd' or a river if $c =$ 'w'.

There is at most one link connecting any two locations.

Constraints

- $2 \leq n \leq 2000$
- $1 \leq m \leq min(4000, n(n-1)/2)$
- $1 \leq x, y \leq n$
- c is either the character `d` or `w`
- No location is connected to more than one river.

Output

A single line with `yes` if a valid duathlon exists on the given map or `no` otherwise.

Input 5	Output 5

```
6 6
1 2 d
2 3 w
2 4 d
3 5 d
4 6 d
5 6 w
```

```
yes
```

Input 6	Output 6

```
6 6
1 2 w
2 3 d
2 4 d
3 5 w
4 6 w
5 6 d
```

```
no
```

Belgium Algorithm Contest

Problem D
Dice chains
Time limit: 3s

An n-dice is a sequence $(v_1, v_2, \ldots, v_n)$ such that $0 \le v_i < n$ and $v_{i-1} \le v_i$. If you roll an n-dice, it will land on each v_i with probability $1/n$. Given two n-dices D_1 and D_2 we say that D_1 is *better* than D_2 if, when both dices are rolled, the probability that D_1 yields a result that is **higher** than the result of D_2 is **larger** than $\frac{1}{2}$.

For example, consider the three 4-dices: $D_1 = (0,0,1,1)$, $D_2 = (0,0,2,2)$ and $D_3 = (3,3,3,3)$. The dice D_1 wins against D_2 with probability $\frac{1}{4}$ (D_1 needs to roll 1 D_2 needs to roll 0) and so D_1 is not better than D_2. However D_2 is also not better than D_1 since it wins agains D_1 with probability $\frac{1}{2}$. On the other hand D_3 is better than both D_1 and D_2.

Your task is simple, given n, build a sequence of **maximum length** $D_1, D_2, \ldots, D_k$ such that D_i is better than D_{i-1} and all D_i's are distinct.

Input

The input contains a single line with one integer n as described above.

Constraints

- $2 \le n \le 6$

Output

If the length of the longest sequence of n-dices is k, you should output $k+1$ lines. The first line should contain the length k. The next k lines should each contain n integers separated by single spaces giving the values of each dice. If there are several solutions any of them will be accepted.

Input 7
2

Output 7
2
0 0
1 1

Belgium Algorithm Contest

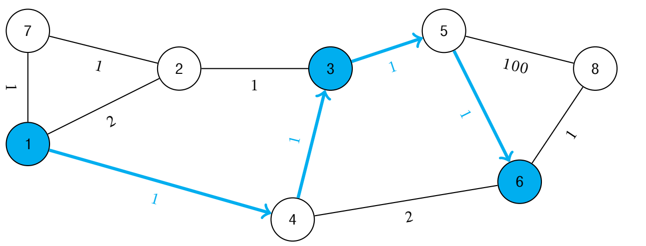
Problem E
Segment Routing
Time limit: 2s

Packets sent over a network follow shortest paths relative to the weights configured on the links. Sometimes it is useful to send traffic on a specific path through the network. This is impossible unless the requested path itself is a shortest path.

Segment routing allows us to overcome this: instead of just specifying the destination, we specify a list of nodes (detours) to visit before the destination. The first node on the list is the origin and the last node is the destination. The packets will then follow the shortest path from the origin to the second node on the list, then from the second to the third and so on until they reach the destination.

Formally, a list of nodes $\ell = \langle x_1, x_2, \ldots, x_k \rangle$ is given and the packets follow a shortest path from x_1 to x_2, then a shortest path from x_2 to x_3 and so on. Note that if several shortest paths exist between two consecutive vertices, there is no way of knowing which path will be used. Therefore, such a list can represent a specific path if and only if there is a unique shortest path between any two consecutive vertices on the list. If ℓ uniquely defines a path, we say that ℓ is *segmentation* of that path and the length of ℓ is the number of elements in the list.

Consider the path $p = (1,4,3,5,6)$ on the graph in the figure. One possible segmentation of p is $\langle 1,4,3,5,6 \rangle$ since each edge of p is the shortest path between its endpoints. However we can do better. The list $\langle 1,3,6 \rangle$ is also a segmentation of p since the unique shortest path from 1 to 3 is $(1,4,3)$ and the unique shortest path from 3 to 6 is $(3,5,6)$. You can check that there is no segmentation of p with a smaller size.



Sometimes a path has no segmentation. For instance, the path $(1,2)$ from the figure has no segmentation because the only candidate is $\langle 1,2 \rangle$ but there are two shortest paths from 1 to 2. Thus it is impossible to uniquely represent $(1,2)$ as a unique sequence of shortest paths.

The link $(5,8)$ belongs to no shortest path due to its cost. Therefore any path containing this link cannot be segmented.

Your taks is to, given a weighted network and a path p, determine the minimum size of a segmentation of p or report that none exists.

Input

The first line of the input contains two integers n and m giving the number of routers and links on the network, respectively.

Then follow m lines each with three integers u, v and w, separated by single spaces, meaning that router u is linked with a bidirectional link of weight w to router v.

The last line contains an integer r and a sequence of integers $p_1, p_2, \ldots, p_r$ separated by single spaces. The first integer, r, represents the number of nodes in the path and the integers $p_1, p_2, \ldots, p_r$ represent the nodes in the path. Note that a path may visit the same node several times. We guarantee that (p_i, p_{i+1}) is a link on the network for each i.

There will be no parallel links in the network.

Constraints

- $2 \leq n \leq 500$
- $1 \leq m \leq n(n-1)/2$
- $1 \leq u < v \leq n$
- $1 \leq w \leq 1000$
- $2 \leq r \leq 10^4$
- $1 \leq p_i \leq n$
- (p_i, p_{i+1}) is a link on the network for each i.

Output

A single line with one integer giving the minimum size of a segmentation of p or `impossible` if p has no segmentation.

Input 8

```
8 11
1 7 1
1 2 2
2 7 1
1 4 1
2 3 1
3 4 1
3 5 1
4 6 2
5 6 1
5 8 100
6 8 1
5 1 4 3 5 6
```

Output 8

```
3
```

Input 9

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

Output 9

```
impossible
```

This page is not really blank.

This page is not really blank.