Lista AA 2022 #3

A. Book of Evil

2 seconds, 256 megabytes

Paladin Manao caught the trail of the ancient Book of Evil in a swampy area. This area contains n settlements numbered from 1 to n. Moving through the swamp is very difficult, so people tramped exactly n - 1 paths. Each of these paths connects some pair of settlements and is bidirectional. Moreover, it is possible to reach any settlement from any other one by traversing one or several paths.

The *distance* between two settlements is the minimum number of paths that have to be crossed to get from one settlement to the other one. Manao knows that the Book of Evil has got a damage range d. This means that if the Book of Evil is located in some settlement, its damage (for example, emergence of ghosts and werewolves) affects other settlements at distance d or less from the settlement where the Book resides.

Manao has heard of m settlements affected by the Book of Evil. Their numbers are $p_1, p_2, ..., p_m$. Note that the Book may be affecting other settlements as well, but this has not been detected yet. Manao wants to determine which settlements may contain the Book. Help him with this difficult task.

Input

The first line contains three space-separated integers n, m and d ($1 \le m \le n \le 100000$; $0 \le d \le n$ - 1). The second line contains m distinct space-separated integers $p_1, p_2, ..., p_m$ ($1 \le p_i \le n$). Then n - 1 lines follow, each line describes a path made in the area. A path is described by a pair of space-separated integers a_i and b_i representing the ends of this path.

Output

Print a single number — the number of settlements that may contain the Book of Evil. It is possible that Manao received some controversial information and there is no settlement that may contain the Book. In such case, print 0.

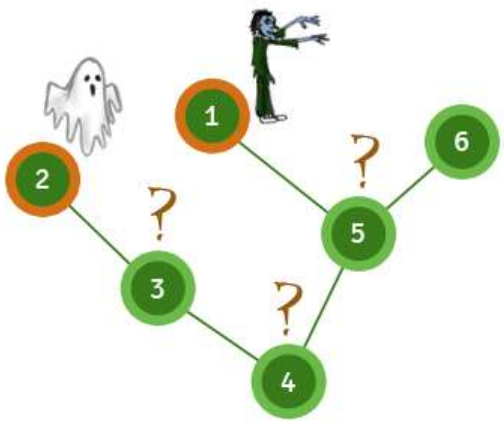
input

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

output

```
3
```

Sample 1. The damage range of the Book of Evil equals 3 and its effects have been noticed in settlements 1 and 2. Thus, it can be in settlements 3, 4 or 5.



B. Treasure Island

1 second, 512 megabytes

All of us love treasures, right? That's why young Vasya is heading for a Treasure Island.

Treasure Island may be represented as a rectangular table $n \times m$ which is surrounded by the ocean. Let us number rows of the field with consecutive integers from 1 to n from top to bottom and columns with consecutive integers from 1 to m from left to right. Denote the cell in r-th row and c-th column as (r, c). Some of the island cells contain impassable forests, and some cells are free and passable. Treasure is hidden in cell (n, m).

Vasya got off the ship in cell $(1, 1)$. Now he wants to reach the treasure. He is hurrying up, so he can move only from cell to the cell in next row (downwards) or next column (rightwards), i.e. from cell (x, y) he can move only to cells $(x + 1, y)$ and $(x, y + 1)$. Of course Vasya can't move through cells with impassable forests.

Evil Witch is aware of Vasya's journey and she is going to prevent him from reaching the treasure. Before Vasya's first move she is able to grow using her evil magic impassable forests in previously free cells. Witch is able to grow a forest in any number of any free cells except cells $(1, 1)$ where Vasya got off his ship and (n, m) where the treasure is hidden.

Help Evil Witch by finding out the minimum number of cells she has to turn into impassable forests so that Vasya is no longer able to reach the treasure.

Input

First line of input contains two positive integers n, m ($3 \le n \cdot m \le 1\,000\,000$), sizes of the island.

Following n lines contains strings s_i of length m describing the island, j-th character of string s_i equals "#" if cell (i, j) contains an impassable forest and "." if the cell is free and passable. Let us remind you that Vasya gets of his ship at the cell $(1, 1)$, i.e. the first cell of the first row, and he wants to reach cell (n, m), i.e. the last cell of the last row.

It's guaranteed, that cells $(1, 1)$ and (n, m) are empty.

Output

Print the only integer k, which is the minimum number of cells Evil Witch has to turn into impassable forest in order to prevent Vasya from reaching the treasure.

input

```
2 2
..
..
```

output

```
2
```

input

```
4 4
....
#.#.
....
.#..
```

output

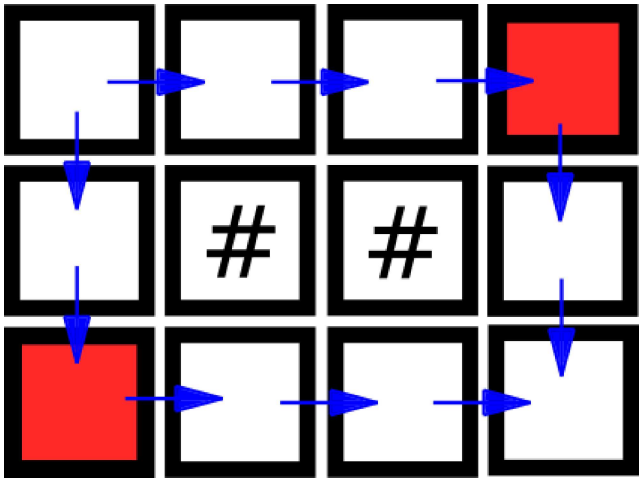
```
1
```

input

```
3 4
....
.##.
....
```

output
2

The following picture illustrates the island in the third example. Blue arrows show possible paths Vasya may use to go from $(1, 1)$ to (n, m). Red illustrates one possible set of cells for the Witch to turn into impassable forest to make Vasya's trip from $(1, 1)$ to (n, m) impossible.



C. Two Small Strings

2 seconds, 256 megabytes

You are given two strings s and t **both of length** 2 and both consisting only of characters 'a', 'b' and 'c'.

Possible examples of strings s and t: "ab", "ca", "bb".

You have to find a string res consisting of $3n$ characters, n characters should be 'a', n characters should be 'b' and n characters should be 'c' and s and t should not occur in res as substrings.

A substring of a string is a contiguous subsequence of that string. So, the strings "ab", "ac" and "cc" are substrings of the string "abacc", but the strings "bc", "aa" and "cb" are not substrings of the string "abacc".

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

Input

The first line of the input contains one integer n $(1 \le n \le 10^5)$ — the number of characters 'a', 'b' and 'c' in the resulting string.

The second line of the input contains one string s of length 2 consisting of characters 'a', 'b' and 'c'.

The third line of the input contains one string t of length 2 consisting of characters 'a', 'b' and 'c'.

Output

If it is impossible to find the suitable string, print "NO" on the first line.

Otherwise print "YES" on the first line and string res on the second line. res should consist of $3n$ characters, n characters should be 'a', n characters should be 'b' and n characters should be 'c' and s and t should not occur in res as substrings.

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

input
2 ab bc
output
YES acbbac

input
3 aa bc
output
YES cacbacbab

input
1 cb ac
output
YES abc

D. Fox And Jumping

2 seconds, 256 megabytes

Fox Ciel is playing a game. In this game there is an infinite long tape with cells indexed by integers (positive, negative and zero). At the beginning she is standing at the cell 0.

There are also n cards, each card has 2 attributes: length l_i and cost c_i. If she pays c_i dollars then she can apply i-th card. After applying i-th card she becomes able to make jumps of length l_i, i. e. from cell x to cell $(x - l_i)$ or cell $(x + l_i)$.

She wants to be able to jump to any cell on the tape (possibly, visiting some intermediate cells). For achieving this goal, she wants to buy some cards, paying as little money as possible.

If this is possible, calculate the minimal cost.

Input

The first line contains an integer n $(1 \le n \le 300)$, number of cards.

The second line contains n numbers l_i $(1 \le l_i \le 10^9)$, the jump lengths of cards.

The third line contains n numbers c_i $(1 \le c_i \le 10^5)$, the costs of cards.

Output

If it is impossible to buy some cards and become able to jump to any cell, output -1. Otherwise output the minimal cost of buying such set of cards.

input
3 100 99 9900 1 1 1
output
2

input
5 10 20 30 40 50 1 1 1 1 1
output
-1

input
7 15015 10010 6006 4290 2730 2310 1 1 1 1 1 1 1 10
output
6

input
8 4264 4921 6321 6984 2316 8432 6120 1026 4264 4921 6321 6984 2316 8432 6120 1026
output
7237

In first sample test, buying one card is not enough: for example, if you buy a card with length 100, you can't jump to any cell whose index is not a multiple of 100. The best way is to buy first and second card, that will make you be able to jump to any cell.

In the second sample test, even if you buy all cards, you can't jump to any cell whose index is not a multiple of 10, so you should output -1.

E. Ehab and the Expected XOR Problem

1 second, 256 megabytes

Given two integers n and x, construct an array that satisfies the following conditions:

- for any element a_i in the array, $1 \le a_i < 2^n$;
- there is no **non-empty** subsegment with bitwise XOR equal to 0 or x,
- its length l should be maximized.

A sequence b is a subsegment of a sequence a if b can be obtained from a by deletion of several (possibly, zero or all) elements from the beginning and several (possibly, zero or all) elements from the end.

Input
The only line contains two integers n and x ($1 \le n \le 18$, $1 \le x < 2^{18}$).

Output
The first line should contain the length of the array l.

If l is positive, the second line should contain l space-separated integers $a_1, a_2, \ldots, a_l$ ($1 \le a_i < 2^n$) — the elements of the array a.

If there are multiple solutions, print any of them.

input
3 5
output
3 6 1 3

input
2 4
output
3 1 3 1

input
1 1
output
0

In the first example, the bitwise XOR of the subsegments are $\{6, 7, 4, 1, 2, 3\}$.

F. Generate a String

2 seconds, 512 megabytes

`zscoder` wants to generate an input file for some programming competition problem.

His input is a string consisting of n letters 'a'. He is too lazy to write a generator so he will manually generate the input in a text editor.

Initially, the text editor is empty. It takes him x seconds to insert or delete a letter 'a' from the text file and y seconds to copy the contents of the entire text file, and duplicate it.

`zscoder` wants to find the minimum amount of time needed for him to create the input file of exactly n letters 'a'. Help him to determine the amount of time needed to generate the input.

Input
The only line contains three integers n, x and y ($1 \le n \le 10^7$, $1 \le x, y \le 10^9$) — the number of letters 'a' in the input file and the parameters from the problem statement.

Output
Print the only integer t — the minimum amount of time needed to generate the input file.

input
8 1 1
output
4

input
8 1 10
output
8

G. Strange Device

1 second, 256 megabytes

This problem is interactive.

We have hidden an array a of n **pairwise different** numbers (this means that no two numbers are equal). You can get some information about this array using a new device you just ordered on Amazon.

This device can answer queries of the following form: in response to the positions of k different elements of the array, it will return the position and value of the m-th among them in the ascending order.

Unfortunately, the instruction for the device was lost during delivery. However, you remember k, but don't remember m. Your task is to find m using queries to this device.

You can ask **not more than n queries.**

Note that the array a and number m are fixed before the start of the interaction and don't depend on your queries. In other words, **interactor is not adaptive.**

Note that you don't have to minimize the number of queries, and you don't need to guess array a. You just have to guess m.

Input
The first line contains two integers n and k ($1 \le k < n \le 500$) — the length of the array and the number of the elements in the query.

It is guaranteed that number m satisfies $1 \le m \le k$, elements $a_1, a_2, \ldots, a_n$ of the array satisfy $0 \le a_i \le 10^9$, and all of them are different.

Interaction
You begin the interaction by reading n and k.

To ask a question about elements on positions $x_1, x_2, \ldots, x_k$, in a separate line output

? x_1 x_2 x_3 ... x_k

Numbers in the query have to satisfy $1 \le x_i \le n$, and all x_i have to be different. Don't forget to 'flush', to get the answer.

In response, you will receive two integers pos and a_{pos} — the position in the array a of the m-th in ascending order element among $a_{x_1}, a_{x_2}, \ldots, a_{x_k}$, and the element on this position.

In case your query is invalid or you asked more than n queries, the program will print -1 and will finish interaction. You will receive a **Wrong answer** verdict. Make sure to exit immediately to avoid getting other verdicts.

When you determine m, output

! m

After printing a query do not forget to output end of line and flush the output. Otherwise, you will get `Idleness limit exceeded`. To do this, use:

- `fflush(stdout)` or `cout.flush()` in C++;
- `System.out.flush()` in Java;
- `flush(output)` in Pascal;
- `stdout.flush()` in Python;
- see documentation for other languages.

Hack format

For the hacks use the following format:

The first line has to contain three integers n, k, m ($1 \le m \le k < n \le 500$) — the length of the array, number of elements in the query, and which in the ascending order number the device returns.

In the next line output n integers $a_1, a_2, \ldots, a_n$ $(0 \le a_i \le 10^9)$ — the elements of the array. They have to be pairwise different.

input
4 3 4 9 4 9 4 9 1 2
output
? 2 3 4 ? 1 3 4 ? 1 2 4 ? 1 2 3 ! 3

In the example, $n = 4$, $k = 3$, $m = 3$, $a = [2, 0, 1, 9]$.

H. Maximal GCD

1 second, 256 megabytes

You are given positive integer number n. You should create such **strictly increasing** sequence of k positive numbers $a_1, a_2, \ldots, a_k$, that their sum is equal to n and greatest common divisor is maximal.

Greatest common divisor of sequence is maximum of such numbers that every element of sequence is divisible by them.

If there is no possible sequence then output `-1`.

Input
The first line consists of two numbers n and k $(1 \le n, k \le 10^{10})$.

Output
If the answer exists then output k numbers — resulting sequence. Otherwise output `-1`. If there are multiple answers, print any of them.

input
6 3
output
1 2 3

input
8 2
output
2 6

input
5 3
output
-1

I. Design Tutorial: Inverse the Problem

2 seconds, 256 megabytes

There is an easy way to obtain a new task from an old one called "Inverse the problem": we give an output of the original task, and ask to generate an input, such that solution to the original problem will produce the output we provided. The hard task of Topcoder Open 2014 Round 2C, InverseRMQ, is a good example.

Now let's create a task this way. We will use the task: you are given a tree, please calculate the distance between any pair of its nodes. Yes, it is very easy, but the inverse version is a bit harder: you are given an $n \times n$ distance matrix. Determine if it is the distance matrix of a weighted tree (all weights must be positive integers).

Input
The first line contains an integer n $(1 \le n \le 2000)$ — the number of nodes in that graph.

Then next n lines each contains n integers $d_{i,j}$ $(0 \le d_{i,j} \le 10^9)$ — the distance between node i and node j.

Output
If there exists such a tree, output "`YES`", otherwise output "`NO`".

input
3 0 2 7 2 0 9 7 9 0
output
YES

input
3 1 2 7 2 0 9 7 9 0
output
NO

input
3 0 2 2 7 0 9 7 9 0
output
NO

input
3 0 1 1 1 0 1 1 1 0
output
NO

input
2 0 0 0 0

output
NO

In the first example, the required tree exists. It has one edge between nodes 1 and 2 with weight 2, another edge between nodes 1 and 3 with weight 7.

In the second example, it is impossible because $d_{1,1}$ should be 0, but it is 1.

In the third example, it is impossible because $d_{1,2}$ should equal $d_{2,1}$.

J. Nastia and a Hidden Permutation

3 seconds, 256 megabytes

This is an interactive problem!

Nastia has a hidden permutation p of length n consisting of integers from 1 to n. You, for some reason, want to figure out the permutation. To do that, you can give her an integer t ($1 \le t \le 2$), two **different** indices i and j ($1 \le i, j \le n, i \ne j$), and an integer x ($1 \le x \le n - 1$).

Depending on t, she will answer:

- $t = 1$: $\max(\min(x, p_i), \min(x + 1, p_j))$;
- $t = 2$: $\min(\max(x, p_i), \max(x + 1, p_j))$.

You can ask Nastia **at most** $\lfloor \frac{3 \cdot n}{2} \rfloor + 30$ times. It is guaranteed that she will **not** change her permutation depending on your queries. Can you guess the permutation?

Input

The input consists of several test cases. In the beginning, you receive the integer T ($1 \le T \le 10\,000$) — the number of test cases.

At the beginning of each test case, you receive an integer n ($3 \le n \le 10^4$) — the length of the permutation p.

It's guaranteed that the permutation is fixed beforehand and that the sum of n in one test doesn't exceed $2 \cdot 10^4$.

Interaction

To ask a question, print "? t i j x" ($t = 1$ or $t = 2$, $1 \le i, j \le n, i \ne j$, $1 \le x \le n - 1$) Then, you should read the answer.

If we answer with -1 instead of a valid answer, that means you exceeded the number of queries or made an invalid query. Exit immediately after receiving -1 and you will see the Wrong Answer verdict. Otherwise, you can get an arbitrary verdict because your solution will continue to read from a closed stream.

To print the answer, print "! p_1 p_2 $\ldots$ p_n (without quotes). **Note that answering doesn't count as one of the $\lfloor \frac{3 \cdot n}{2} \rfloor + 30$ queries.**

After printing a query or printing the answer, do not forget to output end of line and flush the output. Otherwise, you will get `Idleness limit exceeded`. To do this, use:

- `fflush(stdout)` or `cout.flush()` in C++;
- `System.out.flush()` in Java;
- `flush(output)` in Pascal;
- `stdout.flush()` in Python;
- See the documentation for other languages.

Hacks

To hack the solution, use the following test format.

The first line should contain a single integer T ($1 \le T \le 10\,000$) — the number of test cases.

For each test case in the first line print a single integer n ($3 \le n \le 10^4$) — the length of the hidden permutation p.

In the second line print n space-separated integers $p_1, p_2, \ldots, p_n$ ($1 \le p_i \le n$), where p is permutation.

Note that the sum of n over all test cases should not exceed $2 \cdot 10^4$.

input
2 4 3 2 5 3

output
? 2 4 1 3 ? 1 2 4 2 ! 3 1 4 2 ? 2 3 4 2 ! 2 5 3 4 1

Consider the first test case.

The hidden permutation is $[3, 1, 4, 2]$.

We print: "? 2 4 1 3" and get back $\min(\max(3, p_4), \max(4, p_1)) = 3$.

We print: "? 1 2 4 2" and get back $\max(\min(2, p_2), \min(3, p_4)) = 2$.

Consider the second test case.

The hidden permutation is $[2, 5, 3, 4, 1]$.

We print: "? 2 3 4 2" and get back $\min(\max(2, p_3), \max(3, p_4)) = 3$.