A. Points and Segments (easy)

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Iahub isn't well prepared on geometry problems, but he heard that this year there will be a lot of geometry problems on the IOI selection camp. Scared, Iahub locked himself in the basement and started thinking of new problems of this kind. One of them is the following.

Iahub wants to draw n distinct points and m segments on the OX axis. He can draw each point with either red or blue. The drawing is good if and only if the following requirement is met: for each segment $[l_i, r_i]$ consider all the red points belong to it (r_i points), and all the blue points belong to it (b_i points); each segment i should satisfy the inequality $|r_i - b_i| \le 1$.

Iahub thinks that point x belongs to segment $[l, r]$, if inequality $l \le x \le r$ holds.

Iahub gives to you all coordinates of points and segments. Please, help him to find any good drawing.

Input

The first line of input contains two integers: n ($1 \le n \le 100$) and m ($1 \le m \le 100$). The next line contains n space-separated integers $x_1, x_2, ..., x_n$ ($0 \le x_i \le 100$) — the coordinates of the points. The following m lines contain the descriptions of the m segments. Each line contains two integers l_i and r_i ($0 \le l_i \le r_i \le 100$) — the borders of the i-th segment.

It's guaranteed that all the points are distinct.

Output

If there is no good drawing for a given test, output a single integer -1. Otherwise output n integers, each integer must be 0 or 1. The i-th number denotes the color of the i-th point (0 is red, and 1 is blue).

If there are multiple good drawings you can output any of them.

Examples

input

```
3 3
3 7 14
1 5
6 10
11 15
```

output

```
0 0 0
```

input

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

output

```
1 0 1
```

B. Balls Game

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Iahub is training for the IOI. What is a better way to train than playing a Zuma-like game?

There are n balls put in a row. Each ball is colored in one of k colors. Initially the row doesn't contain three or more contiguous balls with the same color. Iahub has a single ball of color x. He can insert his ball at any position in the row (probably, between two other balls). If at any moment there are three or more contiguous balls of the same color in the row, they are destroyed immediately. This rule is applied multiple times, until there are no more sets of 3 or more contiguous balls of the same color.

For example, if Iahub has the row of balls [black, black, white, white, black, black] and a white ball, he can insert the ball between two white balls. Thus three white balls are destroyed, and then four black balls become contiguous, so all four balls are destroyed. The row will not contain any ball in the end, so Iahub can destroy all 6 balls.

Iahub wants to destroy as many balls as possible. You are given the description of the row of balls, and the color of Iahub's ball. Help Iahub train for the IOI by telling him the maximum number of balls from the row he can destroy.

Input

The first line of input contains three integers: n $(1 \leq n \leq 100)$, k $(1 \leq k \leq 100)$ and x $(1 \leq x \leq k)$. The next line contains n space-separated integers $c_1, c_2, \ldots, c_n$ $(1 \leq c_i \leq k)$. Number c_i means that the i-th ball in the row has color c_i.

It is guaranteed that the initial row of balls will never contain three or more contiguous balls of the same color.

Output

Print a single integer — the maximum number of balls Iahub can destroy.

Examples

input

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

output

```
6
```

input

```
1 1 1
1
```

output

```
0
```

C. Xor-tree

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Iahub is very proud of his recent discovery, propagating trees. Right now, he invented a new tree, called xor-tree. After this new revolutionary discovery, he invented a game for kids which uses xor-trees.

The game is played on a tree having n nodes, numbered from 1 to n. Each node i has an initial value $init_i$, which is either 0 or 1. The root of the tree is node 1.

One can perform several (possibly, zero) operations on the tree during the game. The only available type of operation is to pick a node x. Right after someone has picked node x, the value of node x flips, the values of sons of x remain the same, the values of sons of sons of x flips, the values of sons of sons of sons of x remain the same and so on.

The goal of the game is to get each node i to have value $goal_i$, which can also be only 0 or 1. You need to reach the goal of the game by using minimum number of operations.

Input

The first line contains an integer n ($1 \le n \le 10^5$). Each of the next $n - 1$ lines contains two integers u_i and v_i ($1 \le u_i, v_i \le n$; $u_i \ne v_i$) meaning there is an edge between nodes u_i and v_i.

The next line contains n integer numbers, the i-th of them corresponds to $init_i$ ($init_i$ is either 0 or 1). The following line also contains n integer numbers, the i-th number corresponds to $goal_i$ ($goal_i$ is either 0 or 1).

Output

In the first line output an integer number cnt, representing the minimal number of operations you perform. Each of the next cnt lines should contain an integer x_i, representing that you pick a node x_i.

Examples

input

```
10
2 1
3 1
4 2
5 1
6 2
7 5
8 6
9 8
10 5
1 0 1 1 0 1 0 1 0 1
1 0 1 0 0 1 1 1 0 1
```

output

```
2
4
7
```

D. Working out

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Summer is coming! It's time for Iahub and Iahubina to work out, as they both want to look hot at the beach. The gym where they go is a matrix a with n lines and m columns. Let number $a[i][j]$ represents the calories burned by performing workout at the cell of gym in the i-th line and the j-th column.

Iahub starts with workout located at line 1 and column 1. He needs to finish with workout $a[n][m]$. After finishing workout $a[i][j]$, he can go to workout $a[i+1][j]$ or $a[i][j+1]$. Similarly, Iahubina starts with workout $a[n][1]$ and she needs to finish with workout $a[1][m]$. After finishing workout from cell $a[i][j]$, she goes to either $a[i][j+1]$ or $a[i-1][j]$.

There is one additional condition for their training. They have to meet in exactly one cell of gym. At that cell, none of them will work out. They will talk about fast exponentiation (pretty odd small talk) and then both of them will move to the next workout.

If a workout was done by either Iahub or Iahubina, it counts as total gain. Please plan a workout for Iahub and Iahubina such as total gain to be as big as possible. Note, that Iahub and Iahubina can perform workouts with different speed, so the number of cells that they use to reach meet cell may differs.

Input

The first line of the input contains two integers n and m ($3 \le n, m \le 1000$). Each of the next n lines contains m integers: j-th number from i-th line denotes element $a[i][j]$ ($0 \le a[i][j] \le 10^5$).

Output

The output contains a single number — the maximum total gain possible.

Examples

input

```
3 3
100 100 100
100 1 100
100 100 100
```

output

```
800
```

Note

Iahub will choose exercises $a[1][1] \to a[1][2] \to a[2][2] \to a[3][2] \to a[3][3]$. Iahubina will choose exercises $a[3][1] \to a[2][1] \to a[2][2] \to a[2][3] \to a[1][3]$.

E. Guess the Tree

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

Iahub and Iahubina went to a picnic in a forest full of trees. Less than 5 minutes passed before Iahub remembered of trees from programming. Moreover, he invented a new problem and Iahubina has to solve it, otherwise Iahub won't give her the food.

Iahub asks Iahubina: can you build a rooted tree, such that

- each internal node (a node with at least one son) has at least two sons;
- node i has c_i nodes in its subtree?

Iahubina has to guess the tree. Being a smart girl, she realized that it's possible no tree can follow Iahub's restrictions. In this way, Iahub will eat all the food. You need to help Iahubina: determine if there's at least one tree following Iahub's restrictions. **The required tree must contain n nodes**.

Input

The first line of the input contains integer n ($1 \le n \le 24$). Next line contains n positive integers: the i-th number represents c_i ($1 \le c_i \le n$).

Output

Output on the first line "YES" (without quotes) if there exist at least one tree following Iahub's restrictions, otherwise output "NO" (without quotes).

Examples

input

```
4
1 1 1 4
```

output

```
YES
```

input

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

output

```
NO
```