A. Cut 'em all!

1 second, 256 megabytes

You're given a tree with n vertices.

Your task is to determine the maximum possible number of edges that can be removed in such a way that all the remaining connected components will have even size.

Input

The first line contains an integer n ($1 \le n \le 10^5$) denoting the size of the tree.

The next $n-1$ lines contain two integers u, v ($1 \le u, v \le n$) each, describing the vertices connected by the i-th edge.

It's guaranteed that the given edges form a tree.

Output

Output a single integer k — the maximum number of edges that can be removed to leave all connected components with even size, or -1 if it is impossible to remove edges in order to satisfy this property.

input

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

output

```
1
```

input

```
3
1 2
1 3
```

output

```
-1
```

input

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

output

```
4
```

input

```
2
1 2
```

output

```
0
```

In the first example you can remove the edge between vertices 1 and 4. The graph after that will have two connected components with two vertices in each.

In the second example you can't remove edges in such a way that all components have even number of vertices, so the answer is -1.

B. Rumor

2 seconds, 256 megabytes

Vova promised himself that he would never play computer games... But recently Firestorm — a well-known game developing company — published their newest game, World of Farcraft, and it became really popular. Of course, Vova started playing it.

Now he tries to solve a quest. The task is to come to a settlement named Overcity and spread a rumor in it.

Vova knows that there are n characters in Overcity. Some characters are friends to each other, and they share information they got. Also Vova knows that he can bribe each character so he or she starts spreading the rumor; i-th character wants c_i gold in exchange for spreading the rumor. When a character hears the rumor, he tells it to all his friends, and they start spreading the rumor to their friends (for free), and so on.

The quest is finished when all n characters know the rumor. What is the minimum amount of gold Vova needs to spend in order to finish the quest?

Take a look at the notes if you think you haven't understood the problem completely.

Input

The first line contains two integer numbers n and m $(1 \le n \le 10^5, 0 \le m \le 10^5)$ — the number of characters in Overcity and the number of pairs of friends.

The second line contains n integer numbers c_i $(0 \le c_i \le 10^9)$ — the amount of gold i-th character asks to start spreading the rumor.

Then m lines follow, each containing a pair of numbers (x_i, y_i) which represent that characters x_i and y_i are friends $(1 \le x_i, y_i \le n, x_i \ne y_i)$. It is guaranteed that each pair is listed at most once.

Output

Print one number — the minimum amount of gold Vova has to spend in order to finish the quest.

input

```
5 2
2 5 3 4 8
1 4
4 5
```

output

```
10
```

input

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

output

```
55
```

input

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

output

```
15
```

In the first example the best decision is to bribe the first character (he will spread the rumor to fourth character, and the fourth one will spread it to fifth). Also Vova has to bribe the second and the third characters, so they know the rumor.

In the second example Vova has to bribe everyone.

In the third example the optimal decision is to bribe the first, the third, the fifth, the seventh and the ninth characters.

C. Up and Down the Tree

3 seconds, 256 megabytes

You are given a tree with n vertices; its root is vertex 1. Also there is a token, initially placed in the root. You can move the token to other vertices. Let's assume current vertex of token is v, then you make any of the following two possible moves:

- move down to any **leaf** in subtree of v;
- if vertex v is a leaf, then move up to the parent no more than k times. In other words, if $h(v)$ is the depth of vertex v (the depth of the root is 0), then you can move to vertex to such that to is an ancestor of v and $h(v) - k \leq h(to)$.

Consider that root is not a leaf (even if its degree is 1). Calculate the maximum number of different leaves you can visit during one sequence of moves.

Input

The first line contains two integers n and k ($1 \leq k < n \leq 10^6$) — the number of vertices in the tree and the restriction on moving up, respectively.

The second line contains $n - 1$ integers $p_2, p_3, \dots, p_n$, where p_i is the parent of vertex i.

It is guaranteed that the input represents a valid tree, rooted at 1.

Output

Print one integer — the maximum possible number of different leaves you can visit.

input

```
7 1
1 1 3 3 4 4
```

output

```
4
```

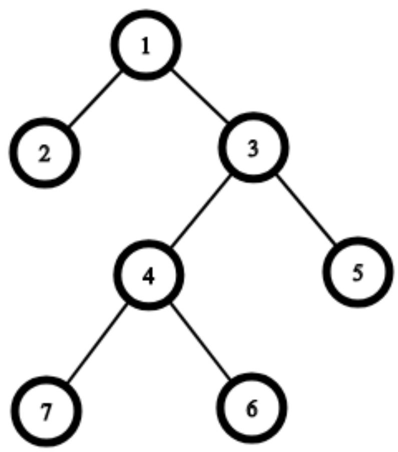
input

```
8 2
1 1 2 3 4 5 5
```

output

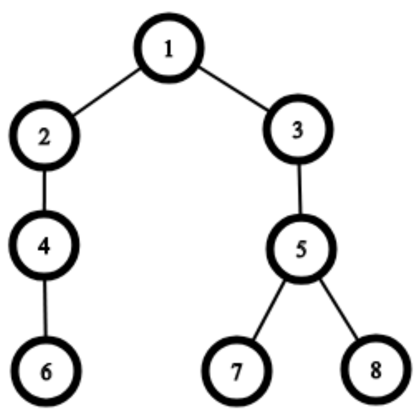
```
2
```

The graph from the first example:



One of the optimal ways is the next one: $1 \to 2 \to 1 \to 5 \to 3 \to 7 \to 4 \to 6$.

The graph from the second example:



One of the optimal ways is the next one: $1 \to 7 \to 5 \to 8$. Note that there is no way to move from 6 to 7 or 8 and vice versa.

D. Almost Acyclic Graph

1 second, 256 megabytes

You are given a directed graph consisting of n vertices and m edges (each edge is directed, so it can be traversed in only one direction). You are allowed to remove at most one edge from it.

Can you make this graph acyclic by removing at most one edge from it? A directed graph is called acyclic iff it doesn't contain any cycle (a non-empty path that starts and ends in the same vertex).

Input

The first line contains two integers n and m ($2 \le n \le 500$, $1 \le m \le min(n(n - 1), 100000)$) — the number of vertices and the number of edges, respectively.

Then m lines follow. Each line contains two integers u and v denoting a directed edge going from vertex u to vertex v ($1 \le u, v \le n$, $u \ne v$). Each ordered pair (u, v) is listed at most once (there is at most one directed edge from u to v).

Output

If it is possible to make this graph acyclic by removing at most one edge, print `YES`. Otherwise, print `NO`.

input

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

output

```
YES
```

input

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

output

```
NO
```

In the first example you can remove edge $2 \rightarrow 3$, and the graph becomes acyclic.

In the second example you have to remove at least two edges (for example, $2 \rightarrow 1$ and $2 \rightarrow 3$) in order to make the graph acyclic.

E. Can of Worms

2 seconds, 256 megabytes

There is an old adage about opening a can of worms. A lesser known adage is one about shooting a can of exploding worms with a BB gun.

Imagine we place some cans of exploding worms on a long, straight fence. When a can is shot, all of the worms inside will explode. Different types of worms have different blast radii. Each can contains only one kind of worm.

When a can explodes, if another can is in the blast radius, then that can will also explode, possibly creating a chain reaction. Each can explodes only once. This process continues until all explosions stop. For each can, suppose that it is the only can shot. How many cans in total will explode?

Input

The input file will begin with a line with a single integer n ($1 \leq n \leq 100,000$) representing the number of cans on that fence. Each of the next n lines will have two integers x ($-10^9 \leq x \leq 10^9$) and r ($1 \leq r \leq 10^9$), where x is the location of the can on the fence and r is the blast radius. No two cans will occupy the same location.

Output

Print n integers on a single line separated by spaces. The ith integer represents the number of cans that will explode if the ith can is the one that is shot.

input

```
3
4 3
-10 9
-2 3
```

output

```
1 2 1
```

input

```
12
2 2
7 7
10 1
19 3
23 12
29 8
33 1
35 17
39 2
40 1
46 11
52 3
```

output

```
1 3 1 1 9 9 1 9 2 2 9 1
```