

Head out to the Target

Input file: **standard input**
Output file: **standard output**
Time limit: 3 seconds
Memory limit: 512 megabytes

You are given a rooted tree with n nodes. The root is node 1. For each node i ($2 \leq i \leq n$), its parent is denoted by f_i , meaning there is an edge between node i and node f_i . A piece is initially located at node 1.

In this problem, we assume that time progresses in discrete moments starting from 1. There are k **non-overlapping** time intervals. During each interval $[l_i, r_i]$, a target appears at node u_i . You are allowed to cut any number of edges in the tree at any moment in time (including multiple times and even at moment 0). Once an edge is cut, it is **permanently removed**.

At any moment:

- If a target is active (i.e., the current moment is within some $[l_i, r_i]$), and
- The piece and the target are in the same connected component, and
- The piece is not currently at the target node,

Then, the piece will move exactly one step along the unique simple path towards the target. If the piece arrives at the target node (or is already at the target node at the beginning of the moment), then a *coincidence* occurs.

Your goal is to determine the earliest possible moment when the piece can *coincide* with any target, possibly by cutting edges strategically. If it is impossible to ever achieve *coincidence* with any target, output -1 .

Input

The first line of input contains two integers n and k ($1 \leq n \leq 10^6$, $1 \leq k \leq 10^6$) — the size of the tree and the number of intervals.

The second line contains $(n - 1)$ integers f_2, \dots, f_n ($1 \leq f_i < i$) — the parents of nodes $2, \dots, n$, respectively.

Then k lines follow, the i -th of which contains three integers u_i, l_i and r_i ($1 \leq u_i \leq n$, $1 \leq l_i \leq r_i \leq 10^9$), indicating that a target appears at node u_i during moments l_i through r_i (inclusive).

It is guaranteed that all time intervals are given in order and do not overlap; that is, for each $1 \leq i < k$, $r_i < l_{i+1}$ holds.

Output

Output a single integer: the minimum moment when the piece can coincide with a target. If it is impossible, output -1 .

Examples

standard input	standard output
8 3 1 1 1 4 4 6 7 5 1 1 8 2 3 4 5 5	5
5 5 1 2 3 3 4 4 6 5 7 7 1 8 8 2 9 9 1 10 10	6
5 1 1 2 3 4 3 1 1	-1

Note

In the first test case, one of the possible optimal strategies is described as follows:

- Moment 1: the piece moves from node 1 to node 4.
 - After the move is made, we cut the edge between nodes 6 and 7.
- Moments 2 and 3: node 4 (the node where the piece is currently at) and node 8 (the target node) are not connected, so the piece does not move.
- Moment 4: There is no target on the tree, so the piece does not move, either.
- Moment 5: The target appears exactly at the node 4 where the piece is located, resulting in a *coincidence*.

It can be easily proven that no strategy allows for a *coincidence to occur before moment 5*. Hence, the answer is 5.