Problem Arboras

Input file: `standard input`
Output file: `standard output`

Roxanne the mage, after countless hours researching ancient arcana, has decided to go to the local cafe to relax. When she arrived at the old cafe, she saw a strange structure on the wall called an *arboras* (or tree). Formally, it's a collection of N vertices numbered with consecutive non-negative integers, where vertex 0 is the root, and all other vertices have a unique parent (vertex v has parent p_v). Since the cafe is run by mages and programmers collectively, the *arboras* (or tree) is drawn with the root at the top.

The mage is intrigued by this structure, and she decides to pour some magic coffee into one of the vertices. If the coffee is poured into vertex u, then it will flow downwards, through the subtree rooted at vertex u. Since it is magic coffee, it doesn't flow randomly: it occupies the *longest chain* it can possibly occupy, within the subtree rooted at u, **while passing through the vertex** u. The amount of coffee lost when pouring is proportional to the length of the chain the coffee occupies. Roxanne denotes this amount as r_u. Note that individual edges of the tree might have different lengths.

Roxanne is interested in the amount of coffee she would lose if she poured it in all the vertices of the tree, that is, the sum of r_u over all vertices u of the tree. This is not difficult to compute at first, but then the programmers decide to challenge her, and **increase** the length of certain edges Q times. Can you help Roxanne find the total length of all the chains occupied by the coffee if poured into all vertices, initially and after each of the Q updates? Beware! She needs the answers **modulo** $10^9 + 7$.

Input

The first line contains integer N, the number of vertices.

The second line contains $N - 1$ integers: $p_1, p_2, \ldots, p_{N-1}$, where p_v is the parent of vertex v, while vertex 0 is the root.

The third line contains $N - 1$ integers: $d_1, d_2, \ldots, d_{N-1}$, where d_v is the length of the edge between vertices v and p_v.

The fourth line contains Q, the number of updates.

Each of the next Q lines contain two integers v_i and add_i, representing the i^{th} update: the length of the edge between vertices v_i and p_{v_i} is increased by add_i.

Output

Print $Q + 1$ lines: on the $i + 1^{\text{th}}$ line you should print the answer after the i^{th} update. On the first line you should print the answer before any updates.

All answers must be printed **modulo** $10^9 + 7$.

Constraints

- $1 \leq N \leq 100\ 000$
- $1 \leq Q \leq 100\ 000$
- $1 \leq d_i \leq 100\ 000\ 000$ for all $1 \leq i \leq N - 1$
- $0 \leq p_i < i$
- $1 \leq add_i \leq 10^9$ for all $1 \leq i \leq Q$

Subtask 1 (11 points)

- $1 \leq N \leq 1\ 000$
- $1 \leq Q \leq 1\ 000$

Subtask 2 (13 points)

- The height of the tree is at most 50.

Subtask 3 (31 points)

- $d_i = 100\ 000\ 000$ for all $1 \leq i \leq N - 1$
- $add_i = 1$ for all $1 \leq i \leq Q$

Subtask 4 (45 points)

- No additional constraints.

Example

input	output
5	0
0 0 1 1	2
0 0 0 0	4
10	8
1 2	10
2 2	12
3 2	13
4 2	14
4 1	15
3 1	2015
2 1	3015
1 1	
4 1000	
2 1000	