Problem F. Flips

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

You are given a rooted tree with n vertices, with vertex 1 as the root. The parent of vertex i ($2 \le i \le n$) is vertex p_i. Each vertex has a value of either 0 or 1 written on it, and initially, vertex i ($1 \le i \le n$) has the value a_i written on it.

Let a *vertex flip* for vertex i be the following operation: if the value written on vertex x_i is 0, change it to 1; if it is 1, change it to 0.

Let a *path flip* for vertex i be the following operation: perform a vertex flip for every vertex on the path from the root (vertex 1) to vertex i (inclusive).

You need to handle q queries. The i-th query ($1 \le i \le q$) is as follows:

- First, perform a vertex flip for vertex x_i.
- After that, find and print the minimum number of path flip operations required to make all vertices have the value 0.

It can be proved that all zeroes can always be achieved in a finite number of path flip operations.

The changes persist between queries. For example, the second query deals with the tree after performing a vertex flip for vertex x_1 and then performing a vertex flip for vertex x_2.

Input

The first line of the input contains a single integer n ($2 \le n \le 2 \cdot 10^5$).

The second line contains $n - 1$ integers p_i. The i-th integer is the parent vertex of the vertex $i + 1$ ($1 \le p_i \le i$).

The third line contains n integers a_i: the initial values written on the vertices ($0 \le a_i \le 1$).

The fourth line contains a single integer q ($1 \le q \le 2 \cdot 10^5$).

Each of the following q lines contains a single integer x_i: the parameter of the i-th query ($1 \le x_i \le n$).

Output

Print q lines. On the i-th line, print the answer to the i-th query.

Example

standard input	*standard output*
4 1 1 3 0 1 1 0 3 2 1 4	2 1 1