

E. May Holidays

time limit per test: 5 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

It's May in Flatland, and there are m days in this month. Despite the fact that May Holidays are canceled long time ago, employees of some software company still have a habit of taking short or long vacations in May.

Of course, not all managers of the company like this. There are n employees in the company that form a tree-like structure of subordination: each employee has a unique integer id i between 1 and n , and each employee with id i (except the head manager whose id is 1) has exactly one direct manager with id p_i . The structure of subordination is not cyclic, i.e. if we start moving from any employee to his direct manager, then we will eventually reach the head manager. We define that an employee u is a subordinate of an employee v , if v is a direct manager of u , or the direct manager of u is a subordinate of v . Let s_i be the number of subordinates the i -th employee has (for example, $s_1 = n - 1$, because all employees except himself are subordinates of the head manager).

Each employee i has a bearing limit of t_i , which is an integer between 0 and s_i . It denotes the maximum number of the subordinates of the i -th employee being on vacation at the same moment that he can bear. If at some moment strictly more than t_i subordinates of the i -th employee are on vacation, and the i -th employee himself is not on a vacation, he becomes *displeased*.

In each of the m days of May exactly one event of the following two types happens: either one employee leaves on a vacation at the beginning of the day, or one employee returns from a vacation in the beginning of the day. You know the sequence of events in the following m days. Your task is to compute for each of the m days the number of displeased employees on that day.

Input

The first line contains two integers n and m ($2 \leq n, m \leq 10^5$) — the number of employees in the company and the number of days in May.

The second line contains $n - 1$ integers p_2, p_3, \dots, p_n ($1 \leq p_i \leq n$), denoting the direct managers of employees.

The third line contains n integers t_1, t_2, \dots, t_n ($0 \leq t_i \leq s_i$), denoting the bearing limits of employees.

The fourth line contains m integers q_1, q_2, \dots, q_m ($1 \leq |q_i| \leq n$, $q_i \neq 0$), denoting the events. If q_i is positive, then the employee with id q_i leaves for a vacation starting from this day, if q_i is negative, then the employee $-q_i$ returns from a vacation starting from this day. In the beginning of May no employee is on vacation. It is guaranteed that if some employee leaves for a vacation, he is not on a vacation at the moment and vice versa.

Output

Print a sequence of m integers a_1, a_2, \dots, a_m , where a_i is the number of displeased employees on the i -th day.

Examples

input
7 8 4 5 1 1 5 5 0 0 0 1 2 0 0 2 6 3 7 -2 4 -3 1
output
1 1 1 2 2 2 1 0

input
5 6 1 2 3 4 4 0 0 1 0 1 5 2 3 -5 -1
output
0 2 1 0 0 0

Note

In the first sample test after employee with id 2 leaves for a vacation at the first day, the head manager with id 1 becomes displeased as he does not want any of his subordinates to go for a vacation. At the fourth day employee with id 5 becomes displeased as his last remaining employee with id 7 leaves for a vacation. At the fifth day employee with id 2 returns from the vacation, but it does not affect the number of displeased employees as the employees 5 and 1 are still displeased. At the sixth day employee with id 3 returns back from the vacation, preventing the employee with id 5 from being displeased and at the last day the head manager with id 1 leaves for a vacation, leaving the company without the displeased people at all.