Aho-Corasick Automaton

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

Bobo had a tree T with $(n+1)$ nodes labeled with $0, 1, \dots, n$ rooted at node 0. Edges were associated with characters.

Let s_i be the concatenation of characters from root to node i. For every i, bobo would like to find f_i such that s_{f_i} was the longest **proper suffix** of s_i.

Note that $s_0 = \epsilon$ (empty string). String u is **proper suffix** of v if and only if there exists a non-empty string w such that $wu = v$.

Input

The first line contains 1 integer n ($1 \le n \le 2 \times 10^5$).

The second line contains n integers $p_1, p_2, \dots, p_n$ where p_i denotes the parent of node i ($0 \le p_i < i$).

The third line contains n integers $c_1, c_2, \dots, c_n$ where c_i indicates that the edge from node p_i to i was associated with the c_i-th character from the alphabet ($1 \le c_i \le n$).

It is guaranteed that $(p_i, c_i) \ne (p_j, c_j)$ for all $i \ne j$.

Output

n integers $f_1, f_2, \dots, f_n$.

Examples

standard input	standard output
2 0 0 1 2	0 0
2 0 1 1 1	0 1