XOR Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	6 seconds
Memory limit:	512 megabytes

You are given a tree with n nodes labelled from 1 to n, the root of which is the node 1, and the node i has a given value a_i for each $i = 1, 2, \ldots, n$.

We define $d(x, y)$ as the number of edges in the shortest path from the node x to the node y, and define a multiset $p(x, k)$ as $\{a_y \mid y \text{ is in the subtree of } x \text{ and } d(x, y) \leq k\}$. Note that here $a_x \in p(x, k)$.

We define the score of any arbitrary set as the sum of squares of XORs of any two numbers. For example, the score of the set $\{1, 1, 2, 3\}$ should be

$$(1 \oplus 1)^2 + (1 \oplus 2)^2 + (1 \oplus 3)^2 + (1 \oplus 2)^2 + (1 \oplus 3)^2 + (2 \oplus 3)^2 = 27$$

where $\oplus$ denotes the bitwise exclusive-or.

Now you are given the parameter k. For each node x you need to compute the score of $p(x, k)$.

Input

The first line of input contains two integers n, k ($1 \leq k \leq n \leq 100000$), the number of nodes of the tree and the parameter described above.

The second line of input contains n integers, the i-th number a_i ($1 \leq a_i \leq 10^9$) is the value of the i-th node.

The third line of input contains $n - 1$ integers, the i-th number f_{i+1} ($1 \leq f_{i+1} \leq i$) is the parent of the $(i + 1)$-th node.

Output

Output n lines, the i-th line contains a single integer, the score of $p(i, k)$. Note that the answer can be extremely large, please output it modulo 2^{64} instead.

Example

standard input	standard output
6 1 4 3 2 4 3 1 1 1 2 2 5	86 98 0 0 4 0