Treerayser

Your favourite game designer, Jacques, has just released his latest creation – *Treerayser*!

As its name suggests, *Treerayser* is played on a tree. The tree has n nodes, numbered from 1 to n, and is rooted at node 1. For every node $i > 1$, there is an edge connecting i to p_i. Each node also has a *value*: node i has value v_i.

To play *Treerayser*, you must select a subset of nodes such that no node has more than k ancestors (including itself) selected. Then, your score is the sum of the values of your selected nodes. Your goal is to maximise your score.

After playing a few rounds of *Treerayser*, you are disappointed that it does not live up to the greatness of your favourite game of his, *Arrayser*. As such, you decide to write a program to play *Treerayser* for you.*

Subtasks & Constraints

For all subtasks:

- $1 \leq k \leq n \leq 500\,000$.
- $1 \leq p_i < i$ for all i.
- $0 \leq v_i \leq 10^9$ for all i.

Additionally:

- For Subtask 1 (5 points), $p_i = i - 1$ for all i.
- For Subtask 2 (11 points), $p_i = i - 1$ or $p_i = 1$ for all i.
- For Subtask 3 (9 points), $v_i \leq 1$ for all i.
- For Subtask 4 (19 points), $v_i \leq 2$ for all i.
- For Subtask 5 (21 points), $k \leq 20$ and $n \leq 100\,000$.
- For Subtask 6 (18 points), $n \leq 100\,000$.
- For Subtask 7 (17 points), no additional constraints.

Input

- The first line of input contains two integers, n and k.
- The second line contains $n - 1$ integers $p_2, \cdots, p_n$.
- The third line contains n integers $v_1, \cdots, v_n$.

Output

Output a single line containing a single integer: the maximum score you could achieve in this game of *Treerayser*.

Sample Input 1

```
10 2
1 1 1 3 5 2 5 5 3
7 7 9 2 7 3 5 5 3 1
```

Sample Output 1

```
35
```

*Of course, you will spend the time saved playing *Arrayser* instead.

Sample Input 2

```
10 3
1 2 3 1 5 6 1 8 9
9 8 1 7 3 8 4 8 2 1
```

Sample Output 2

```
46
```

Sample Input 3

```
10 2
1 1 3 3 2 4 7 4 4
2 1 0 2 1 1 2 1 1 1
```

Sample Output 3

```
9
```

Explanation

In the first sample case, selecting nodes 2, 3, 4, 6, 7, 8, 9, 10 scores $7+9+2+3+5+5+3+1=35$ points, which is the highest possible score.

In the second sample case, selecting nodes 1, 2, 4, 6, 7, 8, 9 scores $9+8+7+8+4+8+2=46$ points, which is the highest possible score.

In the third sample case, selecting nodes 2, 4, 5, 6, 7, 9, 10 scores $1+2+1+1+2+1+1=9$ points, which is the highest possible score.