Competitive Programming SS24

Submit until end of contest

Problem: party (1.0 second timelimit)

After being at home and doing nothing in the past few weeks, you want to have a party tonight and invite some of your colleagues from work to come to your place. As you know, your colleagues are very lazy and many of them will join only digital. Therefore, you can expect the number of people actually showing up being not more than allowed during the time of corona.

When preparing the list of invitations, you would like to maximize the total fun factor of the invited people. However, from experience you know that there is not much fun when your direct boss is also invited. As you want everybody at the party to have fun, you would rather avoid such a situation for any of the invited colleagues.

Assume that job relationships in your company are represented by a rooted tree T. Vertices of the tree are numbered from 1 to n and represent your colleagues. Each colleague c is assigned a nonnegative weight f_c that represents the amount of fun they will contribute if you invite them to the party. Each colleague whose number is greater than or equal to 2 is also assigned a number b_c, their direct boss. The colleague with number 1 is the CEO of your company, so she has no direct boss.

What is the largest cumulative fun factor you can achieve on your party without inviting both a colleague and their direct boss?

Input The input consists of three lines, describing a single test case.

The first line contains an integer n ($1 \leq n \leq 10^5$), the number of colleagues.

The second line contains n integers. The ith integer is f_i ($0 \leq f_i \leq 10^7$), the fun factor of colleague i.

The third line contains $n-1$ integers. The ith integer is b_{i+1} ($1 \leq b_{i+1} < i+1$), the direct boss of colleague $i+1$.

Output Output a single integer: The largest fun factor you can achieve on your party without inviting both a colleague and their direct boss.

Sample input

```
6
3 2 2 5 6 7
1 1 2 2 5
```

Sample output

```
15
```