# Problem E: Company Tug of War

Filename: *war*

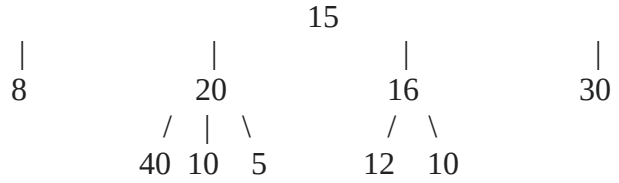
Time limit: *3 seconds*

Leon Skum is a particularly cutthroat CEO. His company has a typical corporate structure where each employee (except for Leon) has an immediate supervisor. All employees who supervise individuals supervise at least 2 individuals directly, but no one supervises more than 10 employees directly, because how on earth could one be productive if they had to supervise more than 10 other employees! Those employees who supervise no one are the peons at the bottom of the food chain.

In order to keep his employees tough, Leon encourages any employee who supervises others in the company to arrange for a tug of war. Naturally, an employee can only compel her subordinates to participate. In particular, if an employee has ***k*** employees directly report to her, he can assign any non-empty subset of those ***k*** employees to one tug of war team, and the rest of those employees by default are on the other team. (Tug of War always has exactly two teams competing.) Once an employee is assigned to a tug of war team, then all of their direct or indirect subordinates must be on that team as well.

To keep these tug of war matches exciting for all involved, all employees who arrange for tug of wars split up the teams so that they are as even as possible. In particular, employee i can pull $w_i$ pounds. A tug of war team's strength is simply the sum of the weights all of the people on the team can pull. Let x be the sum of the weights one team can pull and y be the sum of the weights the other team can pull. The employee arranging for the tug of war will assign teams in such a way |x - y| is minimized. Note that the person arranging for the tug of war match does not participate in it. She gets to watch the festivities for her viewing pleasure.

Let's look at a quick example of one employee arranging a tug of war.




In this example, the employee who can pull weight 15 is organizing the tug of war. She has 4 direct employees, whose subtrees can pull the following sums of weights: 8, 75, 38 and 30, respectively. In order to make the best match, she should put the second subordinate's subtree on one team (weight 75) and the other three subordinates' subtrees on the other team (8 + 38 + 30 = 76). With this arrangement, the teams have a difference of |75 - 76| = 1, which is the minimum achievable amongst all ways to split the teams. (Note: In the input data, the value of 15 for the CEO isn't given because the CEO never competes in any possible tug of war, so this value isn't necessary to solve the problem.)

**The Problem**

Given all of this information, you are curious, what is the maximum difference, |x - y|, amongst all of the tug of war matches that one of the employees might arrange. (Remember, each employee arranges the matches in such a way as to minimize this value.)

**The Input**

The first line of input will contain a single positive integer, $c$ ($1 \leq c \leq 20$), representing the number of input cases to process. Each of the input cases follow.

The first line of each input case contains a single positive integer, $n$ ($3 \leq n \leq 10{,}000$), representing the number of employees in the company. The employees are numbered 0 through $n$-1, with the employee number 0 being the CEO.

The second line of input contains $n$-1 integers, $s_1, s_2, \ldots s_{n-1}$, where $s_i$, represents the number of the direct supervisor of employee i. Each value that appears in this list will appear in between 2 and 10 times, inclusive. Each of these values will be in between 0 and $n$-1, inclusive and no employee will be their own supervisor, and no cyclic supervisory arrangement will occur. (Specifically, the structure denoted by this information will be a tree with the CEO, person 0, at the root.)

The third line of input contains $n$-1 integers, $w_1, w_2, \ldots w_{n-1}$, $(1 \leq w_i \leq 10^6)$, where $w_i$, represents the amount of weight in pounds employee i can pull.

**The Output**

For each test case, on a line by itself, output the maximum difference in team weights between any two teams that might compete in a tug of war, given that each employee who organizes a tug of war attempts to minimize this value.

**Sample Input**

```
2
10
0 0 0 0 2 2 2 3 3
8 20 16 30 40 10 5 12 10
12
10 10 10 11 9 11 9 9 0 0 0
8 30 2 22 25 20 18 16 25 13 18
```

**Sample Output**

```
25
29
```