Algorithm Design Laboratory with Applications

Prof. Stefano Leucci

Problem: *A problem of two trees.*

A computer scientist with green fingers wants to remove a big tree[1] from his garden and has hired a moving company to transport it away. The moving company owns a truck that can carry a maximum weight of $W \in \mathbb{N}^+$.

The tree is too heavy to be transported at once, but it can be cut into smaller (and lighter) pieces. The computer scientist has modelled this problem with a rooted tree[2] $T = (V, E)$ in which each edge $e \in E$ has a weight $w(e) \in \mathbb{N}^+$. The weight $w(T')$ of a subtree T' of T is the sum of the weights $w(e)$ of all edges e in T'.

Cutting a tree T in one of its internal vertices v means splitting T into two $1 + c(v)$ trees $T_0, T_1, \dots, T_{c(v)}$, where $c(v)$ is the number of children $u_1, \dots, u_{c(v)}$ of v in T. In details:

- T_0 is unique tree containing v in the forest obtained by deleting $u_1, \dots, u_{c(v)}$ from T. T_0 is rooted in the same root as T.
- For $i = 1, \dots, c(v)$, T_i is the subtree of T induced by v and all the descendants of u_i in T. T_i is rooted in v.

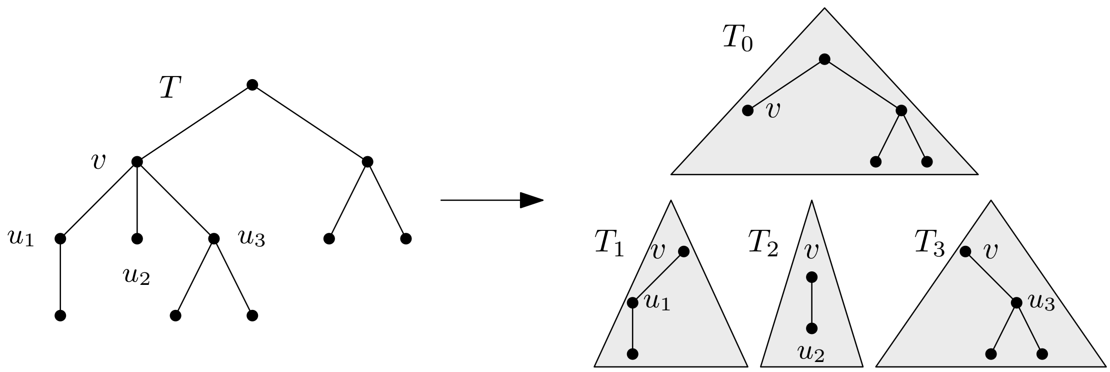


Figure 1: An example of the trees $T_0, \dots, T_{c(v)}$ resulting from cutting T in v. Edge weights are not shown.

Help the computer scientist design a fast algorithm that determines the minimum number $\eta(T, W)$ of cuts needed to decompose T into a forest F in which each tree $T' \in F$ has a weight $w(T')$ of at most W.

Input. The input consists of a set of instances, or *test-cases*, of the previous problem. The first line of the input contains the number C of test-cases. Each test-case is described by 3 lines. The first line of each test-case contains W and the number $n > 0$ of vertices of T. The vertices of T are indexed from 0 to $n - 1$, and the root of T is the vertex with index 0. The second line contains $n - 1$ integers $p_1, \dots, p_{n-1}$ separated by a space, where p_i is the index of the parent (in T) of the unique vertex with index i. The third and final line contains $n - 1$ integers. The i-th of these integers is the weight $w(e)$ of the edge $e = (v, u)$ connecting the vertex u with index i to its parent v in T.

Output. The output consists of C lines. The i-th line is the answer to the i-th test-case and contains the integer η.

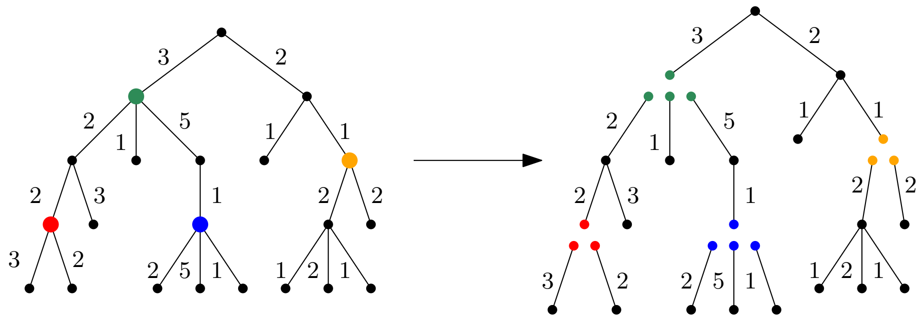
Assumptions. $1 \leq C \leq 10$; $\quad 1 \leq n < 2^{18}$; $\quad \forall e \in E, 1 \leq w(e) \leq 2^{10}$; $\quad \max_{e \in E} w(e) \leq W < 2^{31}$.

[1]Unlike the trees the computer scientist is used to, this tree is made of solid wood and its roots are at the bottom.

[2]The kind of tree the computer scientist is familiar with.

The height of T is at most 2^{10}.
Vertices of T with the same parent have consecutive indices in the input.

Example.



Input (corresponding to the tree in the above figure):

```
1
10 21
0 0 1 1 1 2 2 3 3 5 7 7 8 8 10 10 10 11 11 11
3 2 2 1 5 1 1 2 3 1 2 2 3 2 2 5 1 1 2 1
```

Output (obtained, e.g., by cutting the highlighted vertices):

```
4
```

Requirements. Your algorithm should require $O(n)$ time (with reasonable hidden constants).

Notes. A reasonable implementation should not require more than 0.5 seconds for each input file.