

Jumping Monkey II

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 512 megabytes

There is a tree with n nodes and $n - 1$ edges that make all nodes connected. Each node i has a weight a_i . A monkey is jumping on the tree. In each jump, he can go from one node to another through an edge. If the monkey starts from node u_1 , passes through u_2, u_3, \dots, u_{k-1} , and ends in u_k , where all visited nodes are distinct, we call $u_1, u_2, u_3, \dots, u_k$ a simple path. Let's define the *LIS* (Longest Increasing Subsequence) of a simple path $u_1, u_2, u_3, \dots, u_k$ as follow:

- The longest sequence p_1, p_2, \dots, p_l that meets $1 = p_1 < p_2 < \dots < p_l \leq k$ and $a_{u_{p_1}} < a_{u_{p_2}} < \dots < a_{u_{p_l}}$.

The monkey wants to make the *LIS* as long as possible. For each $i \in [1, n]$, please calculate the maximum length of the *LIS* he can get if he starts from node i .

Input

The first line of the input contains an integer T ($1 \leq T \leq 10^4$), indicating the number of test cases.

The first line of each test case contains an integer n ($1 \leq n \leq 2 \times 10^5$), indicating the number of nodes.

The second line of each test case contains n integers a_1, a_2, \dots, a_n ($1 \leq a_i \leq 10^9$), indicating the weight of each node.

Each of the following $n - 1$ lines contains two integers u, v ($1 \leq u, v \leq n$), indicating an edge connecting nodes u and v . It is guaranteed that the given edges form a tree.

It is guaranteed that $\sum n \leq 2 \times 10^5$ over all test cases.

Output

For each test case, output n lines, the i -th of which is the maximum length of the *LIS* the monkey can get if he starts from node i .

Example

standard input	standard output
2	3
5	2
1 2 3 4 5	2
1 2	2
1 3	1
2 4	2
2 5	2
5	1
3 2 5 4 1	2
1 2	3
2 3	
3 4	
4 5	