Memory, Permutation, and Rooted Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	1024 megabytes

One day, Little P received a tree with n nodes, rooted at node 1, and a permutation p_i of length n. Little P recalled a problem:

Given a rooted tree and a permutation. For each node u in the tree, a node v in the **subtree** of u is called important if and only if $u \neq v$ and the position of v in the permutation appears before that of u. Let d_u be the minimum distance from u to any important node. Specifically, if there are no important nodes for u, then let $d_u = -1$. The distance between two points in the tree is defined as the number of edges in the simple path between them.

After exhausting all efforts to calculate d_i for each node i, Little P set the data aside. As time passed, Little P revisited this problem, where the structure of the rooted tree and the d_i array remained, but the permutation p_i had disappeared. Now, Little P hopes that you can provide a possible permutation p_i from the available information. If there are multiple possible permutations p_i, you need to find the **lexicographically smallest** one.

For two permutations a and b of length n, we say that the lexicographic order of a is less than that of b if and only if there exists $1 \le i \le n$ such that $a_i < b_i$, and for all $1 \le j < i$, $a_j = b_j$.

Input

This problem contains multiple test cases. The first line of input contains an integer T ($1 \le T \le 5 \times 10^4$), representing the number of test cases.

For each test case:

The first line contains an integer n ($2 \le n \le 5 \times 10^5$), representing the number of nodes in the rooted tree.

The second line contains n integers $d_1, d_2, \ldots, d_n$ ($-1 \le d_i \le n$), the meanings of which have been given in the problem.

The next $n-1$ lines each contain two integers u, v ($1 \le u, v \le n, u \neq v$), indicating that there is an edge connecting node u and node v. It is guaranteed that these $n-1$ edges form a tree structure.

It is guaranteed that the total sum of n across all test cases does not exceed 5×10^5.

Output

For each test case, if there exists at least one permutation that satisfies the requirements, output a line with n positive integers representing the lexicographically smallest permutation; otherwise, output a line with `-1`.

Example

standard input	standard output
3 5 -1 1 -1 -1 -1 1 2 2 3 2 4 1 5 10 0 1 1 1 -1 -1 -1 -1 -1 -1 5 3 4 3 8 4 4 2 4 1 2 10 9 5 5 7 6 1 10 2 -1 -1 -1 1 -1 -1 -1 -1 1 2 6 6 10 5 6 5 7 8 10 3 6 5 4 6 9 1 10	1 3 2 4 5 -1 6 1 2 3 4 5 7 8 9 10

Note

For the first test case, the permutation $p = [1, 2, 3, 4, 5]$ is not feasible because, under this permutation, there are no important nodes for node 2, thus $d_2 = -1 \neq 1$. Similarly, the permutation $p = [3, 4, 1, 5, 2]$ is also not feasible because, under this permutation, $d_1 = 2 \neq -1$.