Problem A. Artificial Ascent

Time limit: 5 seconds
Memory limit: 1024 mebibytes

Artificial intelligence is about to surpass human possibilities. Who can still keep up?

— ViPriN, *Artificial Ascent*

You are given a tree with N vertices numbered from 1 to N. For every vertex v, the unique simple path from vertex 1 to vertex v contains at most 40 edges.

Each vertex i initially contains an integer A_i. You are also given a target array B. Each integer from 1 to N appears exactly once in A and exactly once in B. In other words, A and B are permutations of the integers from 1 to N.

It is guaranteed that $A_i \neq B_i$ for at least one index i.

You can perform the following operation:

- Choose a vertex v.
- Let P be the unique simple path from vertex 1 to vertex v.
- Sort the integers written on the vertices of P in ascending order, from vertex 1 to vertex v.

Determine whether it is possible to make the integer written on each vertex i equal to B_i using at most N operations.

If it is possible, find a sequence of at most N operations that achieves this.

Input

Each test contains multiple test cases.

The first line contains an integer t — the number of test cases.

Each test case is described as follows.

The first line contains an integer N — the number of vertices in the tree.

The second line contains N integers $A_1, A_2, \ldots, A_N$, where A_i is the initial value written on vertex i.

The third line contains N integers $B_1, B_2, \ldots, B_N$, where B_i is the target value for vertex i.

Each of the next $N-1$ lines contains two integers u and v, describing an undirected edge between vertices u and v.

Output

For each test case, if it is impossible to achieve the target configuration using at most N operations, print `NO`.

Otherwise, print `YES` on the first line.

On the second line, print an integer k ($1 \leq k \leq N$) — the number of operations.

On the third line, print k integers $v_1, v_2, \ldots, v_k$ ($1 \leq v_j \leq N$), where v_j is the vertex chosen in the j-th operation.

You don't have to minimize the number of operations. If there are multiple valid solutions, you may output any of them.

Constraints

- $1 \leq t \leq 10^5$

- $2 \le N \le 2 \cdot 10^5$
- $1 \le u, v \le N$
- Both A and B are permutations of the integers from 1 to N.
- $A_i \neq B_i$ for at least one i with $1 \le i \le N$.
- The given edges form a tree.
- For every vertex v, the path connecting vertices 1 and v contains at most 40 edges.
- The sum of N over all test cases does not exceed $2 \cdot 10^5$.

Example

standard input	standard output
3 3 3 1 2 1 2 3 1 2 1 3 4 2 3 4 1 1 2 4 3 1 2 2 3 3 4 7 7 4 5 1 3 2 6 1 2 3 4 5 6 7 1 2 1 3 2 4 2 5 3 6 3 7	YES 2 3 2 NO YES 6 3 7 2 5 6 4

Note

In the first test case of the example, the values written on the three vertices are $[3, 1, 2]$. We can achieve the target configuration by performing the following two operations:

1. Choose vertex 3. The path is $1 \to 3$, and the values on this path are $[3, 2]$. After sorting them in ascending order, the values on the three vertices become $[2, 1, 3]$.
2. Choose vertex 2. The path is $1 \to 2$, and the values on this path are $[2, 1]$. After sorting them in ascending order, the values on the three vertices become $[1, 2, 3]$, which is exactly the target configuration.

In the second test case of the example, it is impossible to achieve the target configuration.