Inverted

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Given a tree with n nodes initially numbered from 1 to n, and a node sequence of $n-1$ length, we are going to perform operations on the tree according to the order of the sequence.

For each operation, if the node to be operated is x, firstly create a new node numbered $x+n$. For any integer $i \in [1, n]$ that the edge (x, i) exists:

- If the node $i+n$ does not exist, we connect $(x+n, i)$.
- If the node $i+n$ exists (in this case, the edge $(x, i+n)$ always exists), we connect $(x+n, i+n)$ and delete edge $(x, i+n)$.

For the resulting graph after each operation, calculate the number of spanning trees modulo 998244353.

Input

The first line contains an integer n ($1 \le n \le 5000$), indicating the size of the tree.

The next $n-1$ lines each contain two numbers u and v ($1 \le u, v \le n$), representing an edge (u, v) in the tree. It is guaranteed that the input forms a valid tree.

The next line contains $n-1$ **distinct** numbers b_i ($1 \le b_i \le n$), representing the sequence of nodes to be operated in order.

Output

Output $n-1$ lines, the only number in i-th line represents the number of spanning trees in the graph after the i-th operation, modulo 998244353.

Example

standard input	standard output
5 1 2 1 3 2 4 2 5 1 5 2 3	4 4 6 1