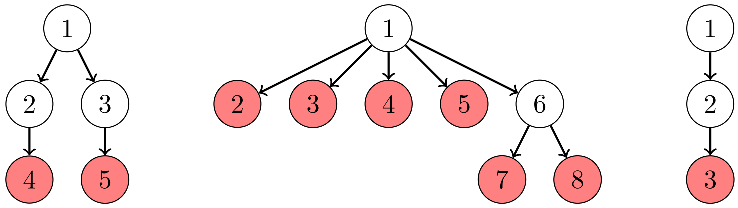
Problem B. Be Careful

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	1024 mebibytes

You are given a rooted tree with n vertices, where the root is vertex 1. A vertex is a *leaf* if it is not the root vertex and its degree is exactly 1.



The figure corresponds to the sample tests, where the leaves are marked red.

Let $\operatorname{mex}(S)$ be the minimal non-negative integer that is not present in S. For example, $\operatorname{mex}\{0, 1, 3, 4\} = 2$, $\operatorname{mex}\{2, 3\} = 0$, $\operatorname{mex} \varnothing = 0$.

Let m be the number of leaves in the given tree. You will perform the following procedure:

1. For every **leaf vertex** u, write any integer from $\{0, 1, 2, \ldots, n\}$ to the vertex u.
2. For every **non-leaf vertex** u, the integer written in u will be the mex of the integers written in all the sons of vertex u.

For example, for the first tree which is described in the figure above, if we write integer 0 to vertex 4 and integer 3 to vertex 5, then:

- The integer written in vertex 2 will be $\operatorname{mex}\{0\} = 1$.
- The integer written in vertex 3 will be $\operatorname{mex}\{3\} = 0$.
- The integer written in vertex 1 will be $\operatorname{mex}\{1, 0\} = 2$.

In total, there are $(n + 1)^m$ ways to fill the tree. You would like to know, for all $k \in \{0, 1, 2, \ldots, n\}$, how many ways are there to fill the tree so that the number written in vertex 1 will be exactly k. Since the numbers can be huge, you only need to output them modulo $998\,244\,353$.

Input

The first line of the input consists of a single integer n $(2 \le n \le 200)$.

Each of the next $n - 1$ lines contains two integers x and y $(1 \le x, y \le n,\ x \ne y)$, indicating that there is an edge between vertices x and y. It is guaranteed that the given graph is a tree.

Output

Output $n + 1$ lines. In the i-th line output a single integer, indicating the answer for $k = i - 1$, modulo $998\,244\,353$.

Examples

standard input	standard output
5 1 2 1 3 2 4 2 5	55 127 34 0 0 0
8 1 2 1 3 1 4 1 5 1 6 6 7 6 8	69632 265534 133905 47790 12636 1944 0 0 0
3 1 2 2 3	1 3 0 0