One Goal

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	512 mebibytes

There are n cities and $n-1$ roads in the Republic of Never. The cities are conveniently numbered from 1 to n. Every road connects two cities and can be traversed in both directions. It is possible to move between any two cities using the roads. The distance $d(u, v)$ is defined as the smallest number of roads one needs to use to move from city u to city v.

There are k friends who are looking to meet. The i-th friend lives in city a_i. For the meeting, the friends will choose city v such that $\sum_{i=1}^{k} d(v, a_i)$ is minimum. If there are several such cities, they will choose the one with the smallest number among them.

Unfortunately, you know just the number of friends but nothing about the cities where they live. Every friend might live in any of the n cities; hence, there are n^k options overall. You would like to find the sum of the numbers of cities the friends will choose in all the n^k options. Output this sum modulo 998 244 353.

Input

The first line of the input contains two integers n and k ($2 \le n, k \le 30\,000$), denoting the number of cities and the number of friends.

Each of the following $n-1$ lines contains two integers u_i and v_i ($1 \le u_i, v_i \le n$; $u_i \ne v_i$), denoting the numbers of the cities connected by the i-th road.

It is guaranteed that it is possible to move between any two cities using the roads.

Output

Display the sum of the numbers of cities the friends will choose in all the n^k options, modulo 998 244 353.

Examples

standard input	*standard output*
3 2 1 2 1 3	12
5 3 2 4 5 3 1 3 3 2	357

Note

In the first example, with three cities and two friends, there are $3^2 = 9$ options to consider. In three of these options, the friends are in the same city and they will clearly choose this city for the meeting. In the other six options, the friends are in different cities and they will choose city 1. The total is $(1+2+3)+6 \cdot 1 = 12$.