

(Adv.) Competitive Programming

Submit until 24.05.2019 13:30, via the [judge interface](#)



Problem: bandits (1 second timelimit)

You are a trader in a small kingdom. To save money during construction, the king designed the road network between towns as a tree. In return he made sure that travel between towns is almost instantly if the traveler is not disturbed. Each spring, the weather priest declares the number of mild days k before the cold winter prevents all lawful citizens from traveling. Traders like you have to plan which cities to visit. A journey $[a_1, a_2, \dots, a_k]$ is a sequence of length k , where a_i is the town you visit during the i -th mild day. Of course, you can stay in a town for longer than a day or visit it multiple times per journey. Consecutive towns in a journey do not even need to be adjacent in the road network. In that case, you always take the shortest path from one town to the next.

Recently ruthless bandits started to ambush innocent traders at certain roads. You call a journey dangerous if bandits roam at least one road you need to travel.

Input The first line contains the number of towns $2 \leq n \leq 10^5$ and the number of mild days $2 \leq k \leq 100$. The following $n - 1$ lines each contain a road. An undirected road is given by the endpoints $1 \leq a, b \leq n$ and 1 if bandits roam the road or 0 otherwise.

Output Out of all n^k possible journeys, how many are dangerous this year? Since this number can be quite large, use your abacus to calculate the result modulo your favorite prime: $10^9 + 7$.

Sample input

```
3 5
1 2 1
2 3 0
```

Sample output

```
210
```

```
4 4
1 2 1
2 3 1
3 4 1
```

```
252
```