Road Maintenance

by nikasvanidze

Problem Submissions Leaderboard Discussions

Byteland has N cities (numbered from 1 to N) and $N-1$ bidirectional roads. A *path* is comprised of 1 or more connected roads. It is guaranteed that there is a path from any city to any other city.

Steven is a road maintenance worker in Byteland. He is required to maintain *exactly* M paths on any given workday. He *cannot* work on the same road twice in one day (so no 2 paths can contain the same 2 roads). Steven can start his workday in any city and, once he has finished maintaining a path, teleport to his next starting city.

Given M, help Steven determine how many different possible $M-$path sets will allow him to perform his maintenance duties. Then print the answer modulo 10^9+7.

Input Format

The first line contains 2 space-separated integers, N (the number of cities) and M (the number of roads to maintain).
Each line i of the $N-1$ subsequent lines contains 2 space-separated integers, A_i B_i, describing a bidirectional road between cities A_i and B_i.

Constraints

- $1 \leq N \leq 10^5$
- $1 \leq M \leq 5$
- $A_i \neq B_i$
- $1 \leq A_i, B_i \leq N$

Output Format

Find the number of different $M-$path sets that will allow Steven to complete M orders, and print the answer $\%\ (10^9+7)$.

Sample Input

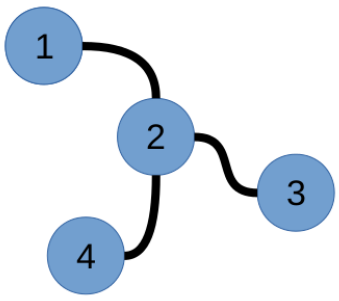
```
4 2
1 2
2 3
2 4
```

Sample Output

```
6
```

Explanation

For the following Byteland map:



Steven can maintain $M=2$ roads using any of the following 6 routes:

1. $[1,2]$ and $[2,3]$
2. $[1,2]$ and $[2,4]$
3. $[1,2]$ and $[3,4]$
4. $[1,3]$ and $[2,4]$

5. $[1, 4]$ and $[2, 3]$
6. $[2, 3]$ and $[2, 4]$

Attempting to reconnect in ... Reconnect

Thus, we print the result of $6 \ \% \ (10^9 + 7)$ on a new line, which is 6.

Submissions: 35
Max Score: 100
Difficulty: Difficult

More

Current Buffer (saved locally, editable) C++

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code