White Falcon was amazed by what she can do with heavy-light decomposition on trees. As a resut, she wants to improve her expertise on heavy-light decomposition. Her teacher gave her an another assignment which requires path updates. As always, White Falcon needs your help with the assignment.

You are given a tree with $N$ nodes and each node's value $val_i$ is initially $0$.

Let's denote the path from node $u$ to node $v$ like this: $p_1, p_2, p_3, \ldots, p_k$, where $p_1 = u$ and $p_k = v$, and $p_i$ and $p_{i+1}$ are connected.

The problem asks you to operate the following two types of queries on the tree:

- "1 u v x" Add $x$ to $val_{p_1}$, $2x$ to $val_{p_2}$, $3x$ to $val_{p_3}$, ..., $kx$ to $val_{p_k}$.
- "2 u v" print the sum of the nodes' values on the path between $u$ and $v$ at modulo $10^9 + 7$.

**Input Format**

First line cosists of two integers $N$ and $Q$ seperated by a space.
Following $N - 1$ lines contains two integers which denote the undirectional edges of the tree.
Following $Q$ lines contains one of the query types described above.

Note: Nodes are numbered by using 0-based indexing.

**Constraints**
$1 \leq N, Q \leq 50000$
$0 \leq x < 10^9 + 7$

**Output Format**

For every query of second type print a single integer.

**Sample Input**

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

**Sample Output**

```
5
```

**Explanation**

After the first type of query, $val_0 = 1, val_1 = 2, val_2 = 3$. Hence the answer of the second query is $2 + 3 = 5$.