F	Tree Weights

A rooted tree with **N** nodes is given. Nodes are labeled 1 to **N**, 1 being the root of the tree. Each of the leaves of this tree has a value assigned to it, which is zero at the beginning. The value for each internal node **U** is calculated as the sum of the values of all the nodes in the sub-tree rooted at **U**. An internal node is a node, which has at least one child node.

You will be given two kinds of operations:

Type 1: given **U**, find the value of node **U**.

Type 2: given **U** and **X**, increase the value of the leaf **U** with **X**.

Input

First line starts with **T (0<T≤10)**, number of test cases. Each of the case starts with **N (0<N≤10^5),** number of nodes in the tree. Next there will be **N-1** lines each containing two integers **U** and **V**, indicating an edge between **U** and **V**. Next there will be **Q (0<Q≤10^5)**, number of operations. Next **Q** line will contain firstly **TP (1 or 2)**, the type of the operation. Then based on the operation type, there will be one or two integers, **U** or **U** and **X (1≤U≤N, |X|≤10^9)**. In case of **TP = 2**, **U** will always be a leaf node.

Output

For each case, print case number. Then for each operation of type 1, print the answer in a separate line. As value of the nodes can get huge, print the answer modulo **1,000,000,007**. See sample I/O for more clarification.

Sample Input	Output for Sample Input
1 4 1 2 1 3 3 4 6 2 2 1 1 1 1 3 2 4 3 1 1 1 3	Case 1: 1 0 7 3

Problem Setter: Hasnain Heickal
Special Thanks: Muhammad Ridowan