

Problem G. Special Christmas Tree

Program: tree.(cpp|java)
Input: tree.in
Balloon Color: White

Christmas is coming and everyone is building a Christmas tree, and you are no exception. However you are special and you want to build a special one. You decided to build a binary tree and to hang its root from the ceiling. For this problem a binary tree can be defined as a collection of connected nodes. The topmost node is called the root. Every node in the tree might have 0, 1 or 2 other nodes hanging from it, called children. Nodes with no children are called leaves. And every node has exactly 1 parent, except the root has no parent.

You bought a decoration pack containing some items and you want to use all of them to decorate all the leaves of the tree. You are limited by the height of your room, so the tree can not be longer than it. The height of tree is the number of edges on the path from the root to the farthest leaf.

Note that each leaf must be decorated by exactly 1 item (and each item can decorate exactly 1 leaf), and you must use all items.

Can you find the most special tree? Tree X is more special than tree Y, if X has more nodes than Y.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer T ($1 \leq T \leq 10,000$) representing the number of test cases. Followed by T test cases. Each test case will consist of a single line, containing 2 integers separated by a single space H and L ($0 \leq H \leq 1,000,000,000$) ($1 \leq L \leq 1,000,000,000$) ($1 \leq L \leq 2^H$) representing the maximum possible height and the number of leaves of the tree, respectively.

Output

For each test case print a single line containing "Case n:" (without quotes) where n is the test case number (starting from 1) followed by a single space then the number of nodes in the most special Christmas tree that has exactly L leaves and height at most H .

Examples

tree.in	stdout
2	Case 1: 7
3 2	Case 2: 9
3 3	