G. Locked Treasure

A group of n $(1 \le n \le 30)$ bandits hid their stolen treasure in a room.   The treasure needs to be locked away until there is a need to retrieve it. Since the bandits do not trust each other, they wanted to ensure that at least k $(1 \le k \le n)$ of the bandits must agree in order to retrieve the treasure.

They have decided to place multiple locks on the door such that the door can be opened if and only if all the locks are opened. Each lock may have up to n keys, distributed to a subset of the bandits. A group of bandits can open a particular lock if and only if someone in the group has a key to that lock.

Given n and k, how many locks are needed such that if the keys to the locks are distributed to the bandits properly, then every group of bandits of size at least k can open all the locks, and no smaller group of bandits can open all the locks?

For example, if $n = 3$ and $k = 2$, only 3 locks are needed — keys to lock 1 can be given to bandits 1 and 2, keys to lock 2 can be given to bandits 1 and 3, and keys to lock 3 can be given to bandits 2 and 3. No single bandit can open all the locks, but any group of 2 bandits can open all the locks.

Input:

The first line of input contains a positive integer indicating the number of cases to follow. Each case is specified by the two integers n and k on one line.

Output:

For each line of input, print on one line the minimum number of locks needed. Follow this format exactly: "`Case`", one space, the case number, a colon and one space, and the answer for that case with no trailing spaces.

Sample Input	Sample Output
2 3 2 5 3	Case 1: 3 Case 2: 10