

Counting Magical Permutatitons

Counting Permutations

In a planet far away from Earth, there is a beautiful country named Magicland. The children of this country play a lot of interesting games with numbers. One of the most popular games is called Inversion. In this game, you will be given numbers from 1 to N . They are given in a certain order. You need to calculate all the inversions in the given permutation of the numbers. S/he who can say it first correctly wins the game. An inversion occurs when there exists a pair of indices i and j such that $i < j$ and given number at i -th position is greater than the number at j -th position.

For example, let us consider a permutation of numbers 1 to 5: 5, 1, 4, 2, 3. This permutation has the following inversions: (5, 1), (5, 4), (5, 2), (5, 3), (4, 2), (4, 3). Therefore, the number of inversion will be 6. The first person to tell this number correctly will win this game.

For this problem, we want to know how many permutations of the numbers 1, 2, ..., N will have at least K inversions.

A permutation X is different from another permutation Y if there exists some i ($1 \leq i \leq N$) for which the number in i -th position is different in these two permutations.

Input

The first line of input file contains the number of test cases, T ($1 \leq T \leq 50$). Then T cases follow:

Each case consists of one line which contains two integers: N and K .

Constraint

For Easy version, $1 \leq N \leq 200$ and $0 \leq K \leq 300$.

For Hard version, $1 \leq N \leq 2000$ and $0 \leq K \leq 3000$.

Output

For each case, print "**Case x : y** " in a separate line, where x is the case number and y is the number of permutations with at least K inversions. As the number can be very large, print **y modulo 10,007**.

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

Problem Setter: Anindya Das