J

Ninja Way

Time Limit	2 seconds
Memory Limit	256 MB

Ninja village is a village where ninja gather. The most important thing is not their physical strength; rather, it's the intel. To have correct information, each village must communicate with each other.

In ninja world, there are **N** ninja villages (**N** is an odd number). Each pair of villages has no more than one road connecting between them and each road is a one-way road. Thus, when consider a pair of villages **A** and **B**, either there is no road connecting them or there is exactly one road. And if that road is heading from **A** to **B**, then there must be no road from **B** directly back to **A**. In that case, however, it is still possible for ninja from village **B** to visit village **A** if there exists a sequence of roads starting from **B** to some other villages (one or more) and then heading to **A**.

We will call a group of villages "**Allied Shinobi Forces**" if it is possible for ninja from each village in the group to visit every other village in this group without walking through other villages outside the group. We will say that "**Allied Shinobi Forces**" has strength **M** if it contains **M** villages in the group.

Write a program to count the maximum number of "**Allied Shinobi Forces**" with strength **M** assuming that the roads are built in an optimal way.

INPUT

The first line of an input will be a positive integer **T** ($T \leq 10^6$) the number of test cases. Each test case is a line containing two integers **N M** ($1 \leq N, M < 10^7$ and **N** is an odd number) where **N** is the total number of ninja villages and **M** is the strength of "**Allied Shinobi Forces**"

OUTPUT

An output of each test case will be a single line "**Case #c: A**" where c is the test case number and **A** is the maximum number of "**Allied Shinobi Forces**" modulo by 10^9+7

EXAMPLE

Sample Input	Sample Output
4 3 3 5 3 5 4 7 3	Case #1: 1 Case #2: 5 Case #3: 5 Case #4: 14