**Learning XOR**

Emily has started learning logic in discrete mathematics and has taken a particular interest in XOR addition. She would like to see how many ways she can use the bitwise XOR operation between a group of four numbers and have them not equal 0. The equation would look like the following:

She would like to take integers **A, B, C, D** and count the number of unique solutions to the problem where she forms a group (**W, X, Y, Z**) where the following rules apply:

* 1 ≤ **W** ≤ **A**
* 1 ≤ **X** ≤ **B**
* 1 ≤ **Y** ≤ **C**
* 1 ≤ **Z** ≤ **D**

When counting the different solutions, two solutions are the same if:

* They contain the same integers
* The number of times each integer occurs in the group is the same

An example of this would be (1, 1, 1, 2) and (1, 1, 2, 1).

**Input:** The first line of input denotes **T** the number of test cases. Each **T** lines after that will contain 4 space-separated integers which denote **A, B, C, D**. It is not guaranteed that they will be in sorted order, and they will be distinct.

**Output:** For each test case, you will output “TEST #(test number): (answer)”, answer being the number of solutions there are.

**Example Input:**

2

1 2 3 4

5 6 7 8

**Example Output:**

CASE #1: 11

CASE #2: 243

**Explanation:**

For case #1, The 11 groups of four are the following: {1, 1, 1, 2}, {1, 1, 1, 3}, {1, 1, 1, 4},

{1, 1, 2, 3}, {1, 1, 2, 4}, {1, 1, 3, 4}, {1, 2, 2, 2}, {1, 2, 2, 3}, {1, 2, 2, 4}, {1, 2, 3, 3}, {1, 2, 3, 4}.

For case #2: The same logic applies, and since numbers can repeat, the maximum that the number of possibilities can reach is 243