New Lottery Game

The Lottery is changing! The Lottery used to have a machine to generate a random winning number. But due to cheating problems, the Lottery has decided to add another machine. The new winning number will be the result of the bitwise-AND operation between the two random numbers generated by the two machines.

To find the bitwise-AND of X and Y, write them both in binary; then a bit in the result in binary has a 1 if the corresponding bits of X and Y were both 1, and a 0 otherwise. In most programming languages, the bitwise-AND of X and Y is written X&Y.

For example:

The old machine generates the number 7 = 0111.

The new machine generates the number 11 = 1011.

The winning number will be (7 AND 11) = (0111 AND 1011) = 0011 = 3.

With this measure, the Lottery expects to reduce the cases of fraudulent claims, but unfortunately an employee from the Lottery company has leaked the following information: the old machine will always generate a non-negative integer less than A and the new one will always generate a non-negative integer less than B.

Catalina wants to win this lottery and to give it a try she decided to buy all non-negative integers less than K.

Given A, B and K, Catalina would like to know in how many different ways the machines can generate a pair of numbers that will make her a winner.

Could you help her?

Input

The first line of the input gives the number of test cases, T. T lines follow, each line with three numbers A B K.

$1 \leq A \leq 10^9$.

$1 \leq B \leq 10^9$.

$1 \leq K \leq 10^9$.

Output

For each test case, output one line containing "Case #x: y", where x is the test case number (starting from 1) and y is the number of possible pairs that the machines can generate to make Catalina a winner.

Example

Input:

```
5
3 4 2
4 5 2
7 8 5
45 56 35
103 143 88
```

Output:

```
Case #1: 10
Case #2: 16
Case #3: 52
Case #4: 2411
Case #5: 14377
```