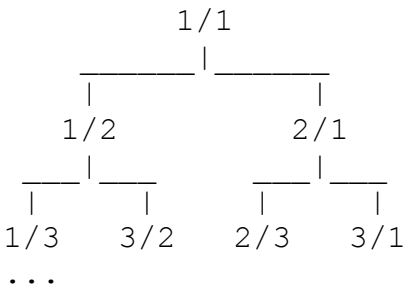
Kick Start 2013 - Round A

Rational Number Tree

Problem

Consider an infinite complete binary tree where the root node is 1/1 and left and right childs of node p/q are p/(p+q) and (p+q)/q, respectively. This tree looks like:



It is known that every positive rational number appears exactly once in this tree. A level-order traversal of the tree results in the following array:

`1/1, 1/2, 2/1, 1/3, 3/2, 2/3, 3/1, ...`

Please solve the following two questions:

1. Find the **n**-th element of the array, where **n** starts from 1. For example, for the input 2, the correct output is 1/2.
2. Given **p/q**, find its position in the array. As an example, the input 1/2 results in the output 2.

Input

The first line of the input gives the number of test cases, **T**. **T** test cases follow. Each test case consists of one line. The line contains a problem id (1 or 2) and one or two additional integers:

1. If the problem id is 1, then only one integer **n** is given, and you are expected to find the **n**-th element of the array.
2. If the problem id is 2, then two integers **p** and **q** are given, and you are expected to find the position of **p/q** in the array.

Output

For each test case:

1. If the problem id is 1, then output one line containing "`Case #x: p q`", where `x` is the case number (starting from 1), and `p`, `q` are numerator and denominator of the asked array element, respectively.
2. If the problem id is 2, then output one line containing "`Case #x: n`", where `x` is the case number (starting from 1), and `n` is the position of the given number.

Limits

Time limit: 30 seconds per test set.
Memory limit: 1GB.

1 ≤ **T** ≤ 100; **p** and **q** are relatively prime.

Test set 1 - Visible

1 ≤ **n**, **p**, **q** ≤ 2^{16}-1; **p**/**q** is an element in a tree with level number ≤ 16.

Test set 2 - Hidden

1 ≤ **n**, **p**, **q** ≤ 2^{64}-1; **p**/**q** is an element in a tree with level number ≤ 64.

Sample

Sample Input

```
4
1 2
2 1 2
1 5
2 3 2
```

Sample Output

```
Case #1: 1 2
Case #2: 2
Case #3: 3 2
Case #4: 5
```