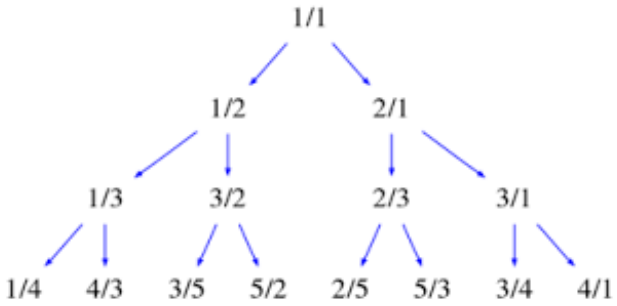
Fractions on Tree (reloaded !)

A fraction tree is an infinite binary tree defined as follows:

1. Every node of tree contains a fraction.
2. Root of tree contains the fraction 1/1.
3. Any node with fraction i/j has two children: left child with fraction i / (i + j) and right child with fraction (i + j) / j.

For example, fraction tree up to 3 levels is as shown:



We number the nodes according to increasing levels (root is at level 1) and at any same level, nodes are numbered from left to right. So first node holds the fraction 1/1, second one holds 1/2, third one holds 2/1 fourth one holds 1/3 and so on.

Your task is simple, as always! Given two numbers a and b, you are to find the product of fractions at all those nodes whose number is between a and b both inclusive.

Input

Every line of the input contains two numbers a and b separated by a space. You are to find the product of all fractions which are at node having number between a and b both inclusive. Input file terminates with a 0 0 which is not to be processed.

Output

For each input, print numerator and denominator of the lowest form of the fraction separated by a /. Output of each case to be on separate lines.

Example

Input:
1 1
1 2
2 4
0 0

Output:
1/1
1/2
1/3

Constraints

1 <= T <= 30000

1 <= a <= b <= 10^10