It’s time to fight the local despots and redistribute the land. There is a rectangular piece of land granted from the government, whose length and width are both in binary form. As the mayor, you must segment the land into multiple squares of equal size for the villagers. What are required is there must be no any waste and each single segmented square land has as large area as possible. The width of the segmented square land is also binary.

**Input**

The first line of the input is T (1 ≤ T ≤ 100), which stands for the number of test cases you need to solve.   
  
Each case contains two binary number represents the length L and the width W of given land. (0 < L, W ≤ 2 1000)

**Output**

For each test case, print a line “Case #t: ”(without quotes, t means the index of the test case) at the beginning. Then one number means the largest width of land that can be divided from input data. And it will be show in binary. Do not have any useless number or space.

**Sample Input**

3

10 100

100 110

10010 1100

**Sample Output**

Case #1: 10

Case #2: 10

Case #3: 110

实际就是求两个数的最大公约数！！！！！！！！！！

//读入两个二进制数，输出这两个二进制数的最大公约数（也是二进制表示）

import java.io.\* ;

import java.math.\* ;

import java.util.\* ;

public class Main

{

public static void main(String[] args)

{

Scanner input=new Scanner(System.in);

int T=input.nextInt();

BigInteger A,B;

for(int i=1;i<=T;i++)

{

A=input.nextBigInteger(2);

B=input.nextBigInteger(2);

System.out.print("Case #"+i+": ");

System.out.println(A.gcd(B).toString(2));

}

}

}