**Large Number Arithmetic**

**Deskripsi Soal:**

Di sebuah desa kecil namun maju secara teknologi bernama Arithmetica, ada kompetisi terkenal yang diadakan setiap tahun dimana pikiran-pikiran paling cerdas berkumpul untuk memecahkan tantangan aritmatika yang kompleks. Tahun ini, hadiah utamanya adalah mesin komputasi canggih yang mampu melakukan perhitungan lebih cepat dari sebelumnya. Tugas ini ditetapkan oleh sesepuh desa, yang dikenal karena kecintaannya pada angka besar dan masalah rumit.

**Format Input:**

Baris pertama input berisi satu bilangan bulat N (1 ≤ N ≤ 100) - jumlah test case.

N baris berikutnya masing-masing berisi dua bilangan bulat a dan b (-1.000.000.000 ≤ a, b ≤ 1.000.000.000) yang mewakili pasangan bilangan tersebut.

**Format Output:**

Untuk setiap pasangan, program harus mencetak lima nilai dalam satu baris:

* Jumlah dari a dan b
* Selisih antara a dan b
* Hasil perkalian dari a dan b
* Hasil pembagian dari a dibagi b (pembagian bilangan bulat)
* Sisa hasil pembagian dari a dibagi b

Jika b adalah nol, baik hasil pembagian maupun sisa harus dicetak sebagai nol.

**Constraints:**

Jumlah test cases N (1 ≤ N ≤ 100)

Dua bilangan bulat a,b (-1.000.000.000 ≤ a, b ≤ 1.000.000.000)

**Sample Input 1 (Standard Input):**

2

2147483647 -2147483647

1000000000 1000000000

**Sample Output 1 (Standard Output):**

0 -2 -4611686014132420609 -1 0

2000000000 0 1000000000000000000 1 0

**Sample Input 2 (Standard Input):**

3

10 0

-15 0

0 0

**Sample Output 2 (Standard Output):**

10 10 0 0 0

-15 -15 0 0 0

0 0 0 0 0

**Penjelasan Case:**

Sample output 1:

Jumlah testcase: 3

Pasangan pertama: 2147483647 dan -2147483647

Jumlah: 2147483647 + (-2147483647) = 0

Selisih: 2147483647 - (-2147483647) = 2147483647 + 2147483647 = -2

Perkalian: 2147483647 \* (-2147483647) = -4611686014132420609

Pembagian: 2147483647 / (-2147483647) = -1

Modulus: 2147483647 % (-2147483647) = 0

Pasangan kedua: 1000000000 dan 1000000000

Jumlah: 1000000000 + 1000000000 = 2000000000

Selisih: 1000000000 - 1000000000 = 0

Perkalian: 1000000000 \* 1000000000 = 1000000000000000000

Pembagian: 1000000000 / 1000000000 = 1

Modulus: 1000000000 % 1000000000 = 0

Sample output 2:

Jumlah testcase: 3

Pasangan pertama: 10 dan 0

Jumlah: 10 + 0 = 10

Selisih: 10 - 0 = 10

Perkalian: 10 \* 0 = 0

Pembagian: 10 / 0 (Division by zero, hasilnya harus 0)

Modulus: 10 % 0 (Modulus by zero, hasilnya harus 0)

Pasangan kedua: -15 dan 0

Jumlah: -15 + 0 = -15

Selisih: -15 - 0 = -15

Perkalian: -15 \* 0 = 0

Pembagian: -15 / 0 (Division by zero, hasilnya harus 0)

Modulus: -15 % 0 (Modulus by zero, hasilnya harus 0)

Pasangan ketiga: 0 dan 0

Jumlah: 0 + 0 = 0

Selisih: 0 - 0 = 0

Perkalian: 0 \* 0 = 0

Pembagian: 0 / 0 (Division by zero, hasilnya harus 0)

Modulus: 0 % 0 (Modulus by zero, hasilnya harus 0)

**Large Number Arithmetic**

**Case Description:**

In a small but technologically advanced village named Arithmetica, there was a well-known competition held every year where the brightest minds gathered to solve complex arithmetic challenges. This year, the grand prize was a state-of-the-art computing machine capable of performing calculations faster than ever before. The task was set by the village's elder, who was known for his fondness of large numbers and intricate problems.

**Input Format:**

The first line of input contains a single integer N (1 ≤ N ≤ 100,000) — the number of test cases.

The next N lines each contain two integers a and b (-1,000,000,000 ≤ a, b ≤ 1,000,000,000) representing the pairs of integers.

**Output Format:**

For each pair, the program should print five values in a single line:

* The sum of a and b
* The difference between a and b
* The product of a and b
* The quotient of a divided by b (integer division)
* The remainder when a is divided by b

If b is zero, both the quotient and remainder should be printed as zero.

**Constraints:**

Number of test cases N (1 ≤ N ≤ 100)

Two integers a, b (-1,000,000,000 ≤ a, b ≤ 1,000,000,000)

**Sample Input 1 (Standard Input):**

2

2147483647 -2147483647

1000000000 1000000000

**Sample Output 1 (Standard Output):**

0 -2 -4611686014132420609 -1 0

2000000000 0 1000000000000000000 1 0

**Sample Input 2 (Standard Input):**

3

10 0

-15 0

0 0

**Sample Output 2 (Standard Output):**

10 10 0 0 0

-15 -15 0 0 0

0 0 0 0 0

**Case Explanation:**

Sample output 1:

Number of test cases : 2

First pair: 2147483647 and -2147483647

Sum: 2147483647 + (-2147483647) = 0

Difference: 2147483647 - (-2147483647) = 2147483647 + 2147483647 = -2

Product: 2147483647 \* (-2147483647) = -4611686014132420609

Division: 2147483647 / (-2147483647) = -1

Modulus: 2147483647 % (-2147483647) = 0

Second pair: 1000000000 and 1000000000

Sum: 1000000000 + 1000000000 = 2000000000

Difference: 1000000000 - 1000000000 = 0

Product: 1000000000 \* 1000000000 = 1000000000000000000

Division: 1000000000 / 1000000000 = 1

Modulus: 1000000000 % 1000000000 = 0

Sample output 2:

Number of test cases : 3

First pair: 10 and 0

Sum: 10 + 0 = 10

Difference: 10 - 0 = 10

Product: 10 \* 0 = 0

Division: 10 / 0 (Division by zero, result should be 0)

Modulus: 10 % 0 (Modulus by zero, result should be 0)

Second pair: -15 and 0

Sum: -15 + 0 = -15

Difference: -15 - 0 = -15

Product: -15 \* 0 = 0

Division: -15 / 0 (Division by zero, result should be 0)

Modulus: -15 % 0 (Modulus by zero, result should be 0)

Third pair: 0 and 0

Sum: 0 + 0 = 0

Difference: 0 - 0 = 0

Product: 0 \* 0 = 0

Division: 0 / 0 (Division by zero, result should be 0)

Modulus: 0 % 0 (Modulus by zero, result should be 0)