

CS1090 PRACTICE PROBLEMS

SEPTEMBER 8, 2015

1. WHICH ORDER?

One often needs to know if a sequence of two numbers is in ascending order, descending order, or they are equal.

INPUT

A sequence of pairs of positive integers less than 1,000,000, two one per line. The sequence -1 -1 marks the end of the input and is not counted.

OUTPUT

One line per input pair, echoing the input values and then indicating whether the pair are in ascending or descending order or are equal.

SAMPLE INPUT	SAMPLE OUTPUT
1 1	1 1 equal
19 21	19 21 ascending
225 224	225 224 descending
401 401	401 401 equal
193765 10	193765 10 descending
-1 -1	

2. GAPS

A sequence of positive integers, in strictly increasing order, may consist of consecutive values, but also may contain gaps. For example, the sequence "3 4 5 9 10 11 12 14 15" has two gaps, one between 5 and 9 (of size 3), the other between 12 and 14 (of size 1). Your task is to find the gaps.

INPUT

The input consists of several increasing sequences (cases) of positive integers less than 1,000,000, with the end marked in each case by a positive value that is less than the preceding one. The values are completely free format with whitespace (spaces, tabs, newlines) separating values. The end of the input is marked by a single negative number. You can assume that a sequence will always have at least one value (plus another less than it that marks the end).

OUTPUT

For each sequence, print the case number, starting at 1, followed by the size of each gap found in the input.

SAMPLE INPUT	SAMPLE OUTPUT
3 4 5 9 10 11 12 14 15 14	Case 1: 3 1
184 183	Case 2:
100 101 200	Case 3: 98 99
300 50	Case 4:
1	
2 3	
1	
-1	

3. PERFECT SQUARES

A perfect square is an integer that is equal to some integer times itself, such as 4, 9, 16, 25, etc. Given an arbitrary positive integer, your task is to find the first perfect square greater than (but not equal to) that integer.

INPUT

The input consists of several positive integers less than 1,000,000, one per line, with a negative value marking the end of the input.

OUTPUT

For each input value, print it and smallest perfect square that is larger than it on one line.

SAMPLE INPUT	SAMPLE OUTPUT
7	7 9
19	19 25
345684	345684 345744
-1	

4. UNIQUENESS

Given a collection of positive values, some of those values only appear once. Your task is to find all the those values that appear only once in the collection (i.e., are unique).

INPUT

The input consists of several positive integers less than 2,000,000,000, with a negative value marking the end of the input. The input is completely free form, with whitespace (spaces, tabs, newlines) separating values.

OUTPUT

Print, in ascending order, the unique values from the input, with commas between each value.

SAMPLE INPUT	SAMPLE OUTPUT
1 1 3 9 8 4 8 2 7 5	2,3,5,7,9
4 1 -1	

5. EVEN NUMBERS

For any pair of non-negative integers, there is some number of even integers (i.e., evenly divisible by 2) between the smallest and the largest. Your task is, for any pair of non-negative integers, to determine how many even numbers that are between them, including those integers if they are even.

INPUT

The input consists of pairs of non-negative (i.e., greater than or equal to 0) integers less than 1,000,000, two per line, with a single negative integer on the last line marking the end.

OUTPUT

Print the two input values followed by the number of even integers between them as shown in the example. If either of the two input integers is even, count it.

SAMPLE INPUT	SAMPLE OUTPUT
1 10	(1,10) 5
10 11	(10,11) 1
56 56	(56,56) 1
0 100	(0,100) 51
-1	