# Problem 2 – Possible Triangles

In Geometry a triangle is real if the **sum of the shortest two sides is larger than the longest side**. Write a program that checks three real numbers if there is a possible triangle to be built from them. You are given lines of **three real numbers.** The last line will hold **“End”.** Print all the possibilities in the format **“a+b>c”**. Where **a** and **b** are the shortest sides and **c** is the longest.

### Input

The input comes from the console. The first lines hold the **three real numbers, separated by single space**. The last line holds the word **“End”**. The input lines will be **distinct** (no duplicates are allowed).

The input data will always be valid and in the format described. There is no need to check it explicitly.

### Output

Print at the console all **possible triangles** found in the input sequence in format "**a+b>c**" (without any spaces), each at a separate line. The **order** of the output lines **should be the same as the order they appear in the input**. The numbers should be printed with **2 digits after the decimal sign**. Print "**No**" in case no triangle is possible among the input sequence of numbers.

### Constraints

* The input **numbers** on each line will be **distinct** real numbers in the range [0…9999].
* There will be no more than 100 lines of input.
* Time limit: 0.5 sec. Memory limit: 16 MB.

### Examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input** | **Output** |  | **Input** | **Output** |
| 20.5 30.6 10.9  1.2345 2.3456 3.4567  1.2 2.3 4.6  End | 10.90+20.50>30.60  1.23+2.35>3.46 | 1 2 3  1 3 4  End | No |

**import** java.math.BigDecimal;

**import** java.util.ArrayList;

**import** java.util.Arrays;

**import** java.util.Scanner;

**import** java.util.TreeSet;

**public** **class** PossibleTriangles {

**public** **static** **void** main(String[] args) {

// Define the initials

Scanner str = **new** Scanner(System.in);

String input = str.nextLine();

ArrayList<String> trinagles = **new** ArrayList<>();

//Read the lines and evaluate them

**while** (!input.equals("End")) {

String[] sides = input.split(" ");

BigDecimal[] numSides = **new** BigDecimal[sides.length];

**for** (**int** i = 0; i < numSides.length; i++) {

numSides[i] = **new** BigDecimal(sides[i]);

}

//sorting the sides and comparing them

Arrays.sort(numSides);

BigDecimal a = numSides[0];

BigDecimal b = numSides[1];

BigDecimal c = numSides[2];

**if** (a.add(b).compareTo(c) > 0) {

trinagles.add(String.format("%.2f+%.2f>%.2f", a, b, c));

}

input = str.nextLine();

}

//Printing out the result if any or No if not

**if** (trinagles.size() > 0) {

**for** (String string : trinagles) {

System.out.println(string);

}

} **else** {

System.out.println("No");