# P1: Using Iterator and ListIterator

******* No points will be given if you violate the rules. *******

## Problem

Write a program using ArrayList<E> to store input double numbers and sort the numbers in ascending order.

## Input

For each test case, input an integer N followed N double numbers. There are many test cases.

## Output

For each case, first use Iterator<E> to print out all numbers in ascending order and then print out all numbers in descending order with ListIterator<E>. Each number（輸出小數點以下兩位）is separated by a space character. Each test case is separated by a newline.

## Sample Input

```
5 2.4 1 4.7 0.5 -2.5 ↲
4 -200 100 12.887 -500.1234↲
```

## Sample Output

```
-2.50 0.50 1.00 2.40 4.70↲
4.70 2.40 1.00 0.50 -2.50↲
↲
-500.12 -200.00 12.89 100.00↲
100.00 12.89 -200.00 -500.12↲
```

# P2 : Binary Search

***** 未依照題目規定，不予計分 *****

## Problem

撰寫一個搜尋程式，此程式必須有以下 generic method：

public static <E extend Comparable<E>> int BinarySearch(E[] list, E key)

請記得作二元搜尋前，必須要先排序。

## Input

輸入有多筆測資，每組測資第一行會輸入資料型態及整數 N 代表有幾筆資料，資料型態分為 Integer、Double、Character 及 String，第二行輸入 N 筆資料。最後會輸入需要搜尋的資料。

## Output

每組測資輸出請先印出排序好之數列，再印出搜尋元素位於 array 的第幾個位置。每組輸出間以空白行隔開。

範例為搜尋 30，有搜尋到請輸出 30 is found in the 3'th place，若找不到則輸出 30 is not found. (double 請輸出至小數點第三位)

| Sample Input | Sample Output |
|---|---|
| String 5↵ | Frank Gill Hank James QQ↵ |
| James Frank Gill Hank QQ↵ | Frag is not found.↵ |
| Frag↵ | ↵ |
| Integer 10↵ | 10 20 30 40 50 60 70 80 90 100 |
| 40 10 100 90 20 60 80 30 70 50↵ | 30 is found in the 3’th place. ↵ |
| Character 8↵ | ↵ |
| E a i o u A U e↵ | A E U a e i o u↵ |
| a↵ | a is found in the 4’th place. ↵ |

# P3: 座標點陣列排序

***** 未依照題目規定不予計分 *****

## Problem

請定義一個整數座標點(Point)類別，此 Point 類別可實作 Comparable 介面，兩點 P1 與 P2 之比較原則如下：

- P1 < P2 if P1 到原點(0,0)的距離（distance）< P2 到原點的距離
- 如 P1 與 P2 到原點的距離相等
  - ✧ P1 < P2 if P1 之 X 座標 ＜ P2 之 X 座標
  - ✧ 如 P1 與 P2 之 X 座標相等，P1 < P2 if P1 之 Y 座標 ＜ P2 之 Y 座標

**Hint: 如 Point 無實作 Comparable 介面，可實作 Comparator 物件，排序時可以使用。**

## Input

本題會有多筆測資，每筆測資先輸入數字 N，代表有 N 個座標點，接著輸入 N 行資料，每行有兩整數分別代表 X 座標與Y座標。

## Output

輸出上述(x, y)由小到大的排序，以及 x+y 相加最大的值與點位 ex: max num: 19518 point(9829,9689)，每組輸出以空白行隔開。(詳細請見 Sample Output)

| Sample Input | Sample Output |
|---|---|
| 3↵<br>1 3↵<br>6 2↵<br>2 6↵<br>6↵<br>5 6↵<br>1 2↵<br>4 3↵<br>2 3↵<br>6 5↵<br>3 4↵ | (1,3)↵<br>(2,6)↵<br>(6,2)↵<br>max num: 8↵<br>point: (6,2)↵<br>↵<br>(1,2)↵<br>(2,3)↵<br>(3,4)↵<br>(4,3)↵<br>(5,6)↵<br>(6,5)↵<br>max num: 11↵<br>point: (6,5)↵ |

# P4: BigDecimal Calculator

## Problem

Write a program to input an expression string in which the operands and operators are separated by zero or more spaces. For example, 3.5*4+3 and 3.5 + 4 % 3 are acceptable expressions. The operator in the expression might be +, -, *, /, and %. Your program must print out the expression and its computing result. The sample output for the input expression 3.5*4+3 is shown below:

3.5 * 4 + 3 = 17

## Requirement

Write a static method BigDecimal calculate(String exp) to compute the expression and return a BigDecimal result. The operands should be stored as BigDecimal in this method. You have to use the arithmetic operators provided by the BigDecimal class to calculate the expression. **(未依規定，以 0 分計)**

## Input

There are many input lines. Each line has an input expression *Exp*. There are three operands and two operators in *Exp*.

## Output

For each input expression *Exp*, please output the expression and its computing result. Note that all tokens are separated by a space character.（小數點以下印一位）

| Sample Input | Sample Output |
|---|---|
| 3 + 4↵ | 3 + 4 = 7.0↵ |
| 32.5-20.5*2↵ | 32.5 - 20.5 * 2 = -8.5↵ |
| 4 * 5.6 + 1.1↵ | 4 * 5.6 + 1.1 = 23.5↵ |
| 20.4 / 4 -3.1↵ | 20.4 / 4 – 3.1 = 2.0↵ |
| 20.8 % 4.1 + 1.8↵ | 20.8 % 4.1 + 1.8 = 2.1↵ |
| -21.5↵ | -21.5↵ |
| 0.0+0.0↵ | 0.0↵ |

# P5: Prime Factorization

## Problem

輸入數字 $N$ (資料型態為 Integer)， 請輸出該數字的所有質因數及其次方。例如 $N=360=2^3*3^2*5$。此題數字可能會有質數出現。

## Requirement

請撰寫以下兩個 static methods：**(未依規定，以 0 分計)**

1. **boolean [] PrimeArray(long N) {….}**
   which returns an array *A* of Boolean values, where *A*[i] is true if i is a prime number, otherwise, *A*[i] is false if i is not a prime number. Note that *A*.length = N+1;
   Hint: if n is a prime number, then n * j is not a prime, where j >= 2;
2. **String PrimeFactorization(long N) {…}**
   which returns a string of prime factorization for the number N. For example, if N = 360, the returned string is "2^3 * 3^2 * 5".

## Input

輸入有多列，每列有個整數 N，最多 1000 列。

## Output

第一行輸出所有數字中之最大數 X 及其開根號整數$\sqrt{X}$，其後針對每一組測資數字 N，輸出 N 的質因數分解，將數字 N 的所有質因數（及其次方）以小到大方式顯示出來，如質因數之次方數大於 1，以^運算符號顯示，不同質因數間以 * 運算符號互相連接， *運算符號前後加空格。

## Sample Input　　Sample Output

| | |
|---|---|
| 360↵<br>3072↵<br>23↵ | 3072 55<br>2^3 * 3^2 * 5↵<br>2^10 * 3↵<br>23↵ |