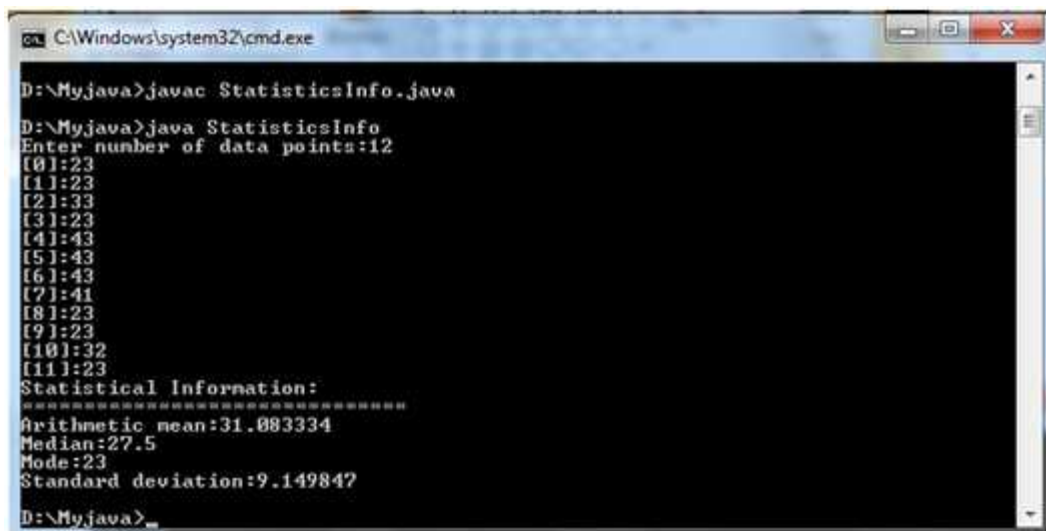


Programming 2

Notice: Your ID and name should be printed before the outputs of all programs.

1. Write a Java program to answer about the statistical information such as arithmetic mean, median, mode, and standard deviation of an integer data set. (The arithmetic median, median, and mode mean that the arithmetic average, mid-point of the distribution, and most frequent response, respectively.) The input data can be generated by random method or keyboard. Your program should display the output similar to the one shown below.



```
C:\Windows\system32\cmd.exe

D:\Myjava>javac StatisticsInfo.java

D:\Myjava>java StatisticsInfo
Enter number of data points:12
[0]:23
[1]:23
[2]:33
[3]:23
[4]:43
[5]:43
[6]:43
[7]:41
[8]:23
[9]:23
[10]:32
[11]:23
Statistical Information:
Arithmetic mean:31.083334
Median:27.5
Mode:23
Standard deviation:9.149847

D:\Myjava>
```

2. Write a program that adds the arguments and return the result as follows. You should use the concept of overloading in your program.

add(1, 2) → 3

add(1, 2, 3) → 6

add(1.4, 4.5) → 5.9

3. Write a program that will read a sequence of positive real numbers entered by the user and will print the same numbers in sorted order from smallest to largest. The user will input a zero to mark the end of the input. Assume that at most 100 positive numbers will be entered.
4. Design a class hierarchy that simulates the things you have. For example, you have many electric devices such as laptop computer, smartphone, and mp3 player, many clothes such as pants, skirts, and jumper, etc. Your design should include abstract classes. Write a program that prints class name, instance variables, and methods of all classes. (Note: Draw your class hierarchy in your document.)
5. Write a program that uses the methods of ArrayList. This program prints 10 numbers randomly generated between 0 to 20. Your program should provide insert, remove, and search operations. Note that numbers should be sorted after all operations are finished. The following is an example.

List: 1 2 4 5 6 8 10 11 13 14

Input your command : a

Input number to add: 7

List: 1 2 4 5 6 7 8 10 11 13 14

Input your command : r

Input number to add: 1

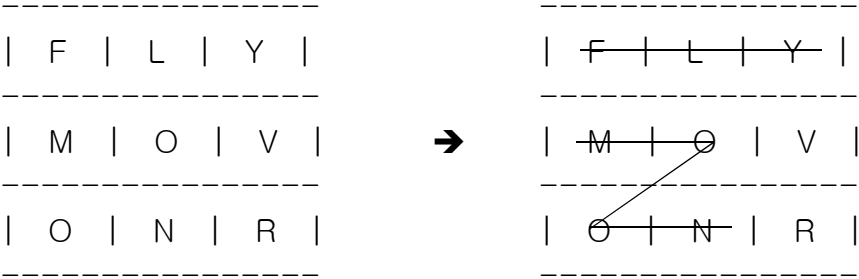
List: 2 4 5 6 8 10 11 13 14

Input your command : s

Input number to search: 5

Index of 5 is 2.

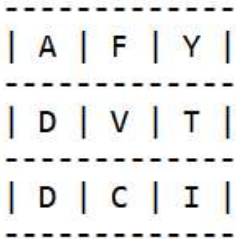
6. Boggle is a simple game that finds words sequentially linear-connected in a NxN grid. In the following example, you can find 2 words the "FLY", "MOON".



Design a light-weight boggle game that satisfies the following requirements.

- 1) Make a Boggle with a 3x3 grid.
- 2) Letters are randomly allocated.
- 3) User may input the word. Find this word at the grid.
- 4) Each word is at least three letters.

The following is an output example.



Input word: DCI
Exist
Input word: ASCT
Non-exist
Input word: |