

LAB EXERCISE

Stats

Background:

1. You will download and save a text file, (*numbers.txt*), containing a large ($N \leq 1000$) number of integers. The integers range in value from 0 to 100. The text file has been created with one value on each line. Due to the potential for the sum of the numbers to be very large, you should use a long integer in your calculation to find the average.
2. The number of integers in the file is unknown. You must read the text file until the EOF marker is encountered.
3. Your program must find the average, standard deviation, and mode of the list of numbers. The mode is defined as the value(s) present with the highest frequency. Calculating the standard deviation consists of the following steps:
 - a. Find the average of the list of numbers.
 - b. Determine the difference of each number from the average, and square each difference. Sum all the differences.
 - c. Divide this sum by (the number of values - 1).
 - d. Take the square root of the above division problem from step c.

Example, given this list of numbers: 7 4 5 9 10

- a. The average = 7
- b. Sum of square of differences:

$$\begin{array}{ccccccccc} (7-7)^2 & + & (4-7)^2 & + & (5-7)^2 & + & (9-7)^2 & + & (10-7)^2 \\ 0 & + & 9 & + & 4 & + & 4 & + & 9 & = 26 \end{array}$$

c. = 6.50

d. = 2.55

4. For a normal distribution, 68.3% of the data will lie within one standard deviation of the average, while 95.4% will lie within two standard deviations.

Assignment:

1. Your program should print out the average, standard deviation, and mode of the data in *numbers.txt*. Format the real numbers to print with 2 decimal places.
2. Your program must utilize proper modular design and parameter passing.
3. Once completed, run your output and upload your source code to Google Classroom.

```
import java.io.File;
```

```

import java.io.FileNotFoundException;
import java.util.Scanner;
public class Stats {
    private int[] myData = new int[1000];
    private int myHowMany;

    public Stats(String fileName)
    {
        myHowMany = 0;
        loadFile(fileName);
    }
    public void loadFile (String fileName)
    /* Loads text file numbers.txt into vector data.  Values are stored from
    positions 0..howMany-1.  Returns the number of data as howMany */
    {
        int index = 0;
        File file = new File(fileName);
        try {
            Scanner inFile = new Scanner(file);

            while (inFile.hasNext())
            {
                myData[index] = inFile.nextInt();
                index++;
            }
        }
        catch (FileNotFoundException e) {
            //e.printStackTrace();
            System.out.println("File is in the wrong directory");
        }
    }

    public double average ()
    // returns average of values in vector data
    {
        return -999;
    }

    public double stdDev ()
    // finds standard deviation of values in vector data
    {
        return -999;
    }
    private int findLargest (int[] scores)
    /* finds the largest integer in vector scores.  This vector is sized from
    0..100, with each element storing the quantity of each number from
    0..100. */
    {
        return -999;
    }
    public void findMode ()
    /* The vector data is analyzed and transferred into a smaller vector
    smallList (0..100).  For each occurrence of n in vector data,
    smallList[n] is incremented +1.  function Largest is then called
    to find the largest quantity in vector smallList.  The mode(s)
    is/are printed out. */

```

```
{  
System.out.println("Print a chart here");  
}
```

```
}
```