

1-D Array LAB TASK

Programming Fundamentals (Java)

Registration # Fa20-Bse-094

Muhammad Ruslan Babar

Submitted to: Sir Azfar Shakeel Khan.

**Question 7.11**

(Statistics: compute deviation) Programming Exercise 5.45 computes the standard deviation of numbers. This exercise uses a different but equivalent formula to compute the standard deviation of n numbers.

To compute the standard deviation with the given formula, you have to store the individual numbers using an array, so that they can be used after the mean is obtained.

**Source Code:**

package Russi7kd;

import java.util.\*;

public class States\_Computer\_Deviation {

public static void main(String[] args) {

//Scanner object

Scanner input = new Scanner(System.in);

System.out.print("\nEnter ten values: ");

//Array declaration

double[] list = new double[10];

//Input data in array

for (int i = 0; i < list.length ; i++)

list[i] = input.nextDouble();

//Method calling and displaying results

System.out.printf("\nThe mean is: %.2f",mean(list));

//Method calling and displaying results

System.out.printf("\nThe deviation is: %.2f",deviation(list));

}

/\*\* Compute the deviation of double values \*/

public static double deviation(double[] x){

double deviation\_Data = 0;

double mean = mean(x);

for (int i = 0; i < x.length; i++)

//Formula

deviation\_Data += Math.pow((x[i] - mean),2) / (x.length - 1);

return Math.sqrt(deviation\_Data);

}

/\*\* Compute the mean of an array of double values \*/

public static double mean(double[] x){

double meanData = 0;

for (int i = 0; i < x.length ; i++)

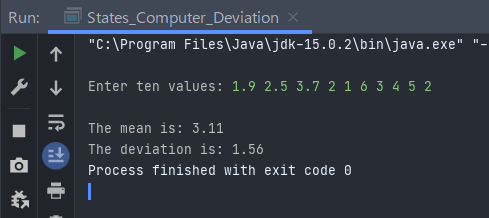
meanData += x[i];

return meanData / x.length;

}

}

**OUTPUT:**



**Question 2**

Improve Listing 7.9 Calculator.java given on Book page 274 to apply same operation on 'N' number of argumnets, e.g. user can enetr 2+34+9+7+6+10

(The code given in the book is aplicable on 2-Args).

**Source Code:**

package Russi7kd;

import java.util.\*;

public class Calculator {

public static void main(String[] args) {

double results = 0;

if (args.length >= 3) {

results = data\_Manupulation(args);

}

else if (args.length < 3){

System.out.println("Enter at least three entries");

System.exit(0);

}

// Displaying Results

System.out.println("\nThe Results are: ");

for (String i: args)

System.out.print(i + " ");

System.out.print("= "+results);

}

// Method for Data Manipulation

public static double data\_Manupulation(String[] string){

double data = 0;

double dataMultiplication = 1;

//===========Addition==============

if (string[1].equals("+")){

for (String i : string)

if (!(i .equals("+")))

data += Double.parseDouble(i);

return data;

}

//============Subtraction=============

else if (string[1].equals("-")){

data = Double.parseDouble(string[0]);

for (int i = 1; i < string.length ; i++) {

if (!(string[i].equals("-"))){

data = data - Math.abs(Integer.parseInt(string[i]));

}

}

return data;

}

//=============Multiplication============

else if (string[1].equals(".") || string[1].equals("\*")) {

for (String k : string) {

if (!(k.equals(".") || k.equals("\*"))){

dataMultiplication \*= Double.parseDouble(k);

}

}

return dataMultiplication;

}

//=============Division============

else if (string[1].equals("/")){

data = Double.parseDouble(string[0]);

for (int i = 1; i < string.length ; i++) {

if (!(string[i].equals("/"))){

data /= Math.abs(Integer.parseInt(string[i]));

}

}

return data;

}

return data;

}

}

**OutPut:**

