

Introducción a Heroku

Student name: *Michael Jefferson Ballesteros Coca*

Course: *Arquitecturas Empresariales* – Professor: *Luis Daniel Benavides Navarro*
Due date: *Agosto 20, 2020*

Program Requirements.

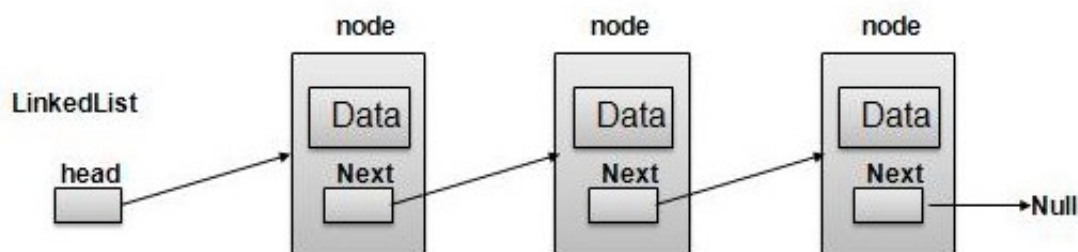
- (a) The program reads n real numbers from a web page
- (b) Test the program with tables 1 and 2
- (c) Use a LinkedList to store the n numbers for calculations.

General.

LinkedList. LinkedLists are common data structures where data is stored, they are implemented with pointers, which allow the data that is used to be chained from memory addresses. LinkedLists have 2 main components:

- List head
- Nodes

There are several options to create this structure, for this repository, the LinkedList has a head, where it has 2 pointers, one towards the beginning of the LinkedList and another towards the last node of the structure.



It also has different functionalities that were implemented, among them:

- add ()
- remove ()
- size ()
- toArray ()

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n a_i = \frac{a_1 + a_2 + a_3 + \dots + a_n}{n}$$

Media.

When we look for the mean of a set of data, we locate the position of the center of these through the average of these.

The Calculator class in the repository has a method (calculateMean ())

```
public static Double calculateMean (Double [] array) {
    Double sum = 0d;
    int n = array.length;
    for (Double x: array) {
        sum += x;
    }
    return sum / n;
}
```

Where, a For loop was used to traverse the data and a variable (sum) is carried to perform the sum of the data; It is concluded after going through the data, the mean, calculating with the value n the amount of data that exist and the sum of all the data.

$$S = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{n - 1}}$$

Standard Deviation.

The standard deviation quantifies the variation of the population, although it would appear more robust with respect to the average if calculated with pencil and paper, it could be a complex procedure.

A method was implemented in the Calculator class to calculate the standard deviation (calculateDeviation ())

```
public static Double calculateDeviation (Double [] array, double mean) {
    Double sumx = 0d;
    int n = array.length;
    for (Double x: array) {
        sumx += Math.pow (x-mean, 2);
    }
    return Math.sqrt ((sumx / (n-1)));
}
```

This time, we use the For loop to determine the internal part of the root, where a variable (sumax) was carried where the calculation was made between the average and each piece of data in the data set. The method ends by returning the square root of the inner part of the root, using the Java Math library.

$$\text{Absolute Error} = |v_A - v_E|$$

Absolute Error.

In order to check our calculations, we need to use the absolute error and have a tolerance margin, since with this we can tell if the resulting data is reliable or not.

```
private static Double TOLERANCE = 0.1d;

@Test
public void shouldCalculateMean () {
    String file = "src \\ test \\ resources \\ data \\ data1.txt";
    LinkedList list = new LinkedList ();
    Calculator.readFile (file, list);
    Double errorAbsoluto = Math.abs (Calculator.calculateMean (list.toArray ()) -
                                     550.6);
    assertTrue (absoluteError <TOLERANCE);
}
```

For this test case, the variable (TOLERANCE) has previously been declared and then compared with the absolute error. We use the data1.txt file that has a set of data to operate, we save this data in the LinkedList to later determine the average of this data. In the variable (errorAbsoluto) the Math library is in charge of giving us the absolute value between the value that was calculated through the application and the number that it was previously supposed to give, after doing the calculation. With a tolerance of 0.1, we use the variable (absoluteError) to determine if the error is less than the tolerance we have.

Web. Using the SparkWebApp class we display a web page, in which we can put data and then send it and the answer is the average and standard deviation

Spark was used, a lightweight container to be able to use basic and not very complex methods to give functionality to the web page.

During the process, the page collects the data and processes it by calling the Calculator object to be able to calculate the average and standard deviation with the values. Lambda functions are used, which allow handling light response and methods for the page that is implemented.

Conclusion. The development was successful, we were able to determine that the procedure for calculating the standard deviation and the average were conclusive and the deployment of the web page met the established criteria.