
***Instituto Tecnológico y de Estudios Superiores de Monterrey***

***Pruebas de software y aseguramiento de la Calidad***

*Titular: Dr. Gerardo Padilla Zárate*

*Asistente: Yetnalezi Quintas Quiroz*

**Ejercicio de programación 1**

A01794283 Francisco José Arellano Montes

Febrero 4, 2024

Todos los resultados completos y el programa se adjuntaron en el repositorio de **Github**:

https://github.com/francisco-arellano/Pruebas-de-software-y-aseguramiento-de-calidad/tree/main/Semana%204

## Actividad 4.2. Ejercicio de programación 1

| Programming Exercise | Description | Practice | Test Cases and Evidence |
|---|---|---|---|
| **1. Compute statistics** | Req1. The program shall be invoked from a command line. The program shall receive a file as parameter. The file will contain a list of items (presumable numbers).<br><br>Req 2. The program shall compute all descriptive statistics from a file containing numbers. The results shall be print on a screen and on a file named *StatisticsResults.txt*. **All computation MUST be calculated using the basic algorithms, not functions or libraries.**<br><br>The descriptive statistics are mean, median, mode, standard deviation, and variance.<br><br>Req 3. The program shall include the mechanism to handle invalid data in the file. Errors should be displayed in the console and the execution must continue.<br><br>Req 4. The name of the program shall be computeStatistics.py<br><br>Req 5. The minimum format to invoke the program shall be as follows:<br>python computeStatistics.py fileWithData.txt<br><br>Req 6. The program shall manage files having from hundreds of items to thousands of items.<br><br>Req 7. The program should include at the end of the execution the time elapsed for the execution and calculus of the data. This number shall be included in the results file and on the screen.<br><br>Req 8. Be compliant with PEP8. | • Control structures<br>• Console Input output<br>• Mathematical computation<br>• File management<br>• Error handling | Record the execution. Use files included in the assignment. |

# Respuestas de ejercicio 1:

**TC1**

The mean is: 242.32

The median is: 239.5

The mode is: 393

The variance is: 21099.917599999997

The SD is: 145.25810683056557

Execution time:0.004003763198852539 seconds

**TC2**

The mean is: 250.7840161861406

The median is: 247

The mode is: 230

The variance is: 20785.369132479238

The SD is: 144.17131868884059

Execution time:0.003999471664428711 seconds

**TC3**

The mean is: 249.77621989860583

The median is: 249.0

The mode is: 94

The variance is: 21117.27747316329

The SD is: 145.31784980917962

Execution time:0.01605963706970215 seconds

**TC4**

The mean is: 149.00267347908746

The median is: 147.75

The mode is: 123.75

The variance is: 17007.920843018837

The SD is: 130.41441961308894

Execution time:0.01596212387084961 seconds

**TC5**

The mean is: 241.49511400651465

The median is: 241.0

The mode is: 466.0

The variance is: 21160.021963097748

The SD is: 145.46484786056646

Execution time:0.0033197402954101562 seconds

**TC6**

The mean is: 1.8790659927977473e+20

The median is: 1.88008049965543e+20

The mode is: NA

The variance is: 1.1530904699530647e+40

The SD is: 1.0738205017381e+20

Execution time:0.0059909820556640625 seconds

**TC7**

The mean is: 2.474673954997149e+20

The median is: 2.4664097307429e+20

The mode is: NA

The variance is: 2.0910793147136484e+40

The SD is: 1.4460564700984703e+20

Execution time:0.021001100540161133 seconds

| Programming Exercise | Description | Practice | Test Cases and Evidence |
|---|---|---|---|
| **2. Converter** | Req1. The program shall be invoked from a command line. The program shall receive a file as parameter. The file will contain a list of items (presumable numbers).<br><br>Req 2. The program shall convert the numbers to binary and hexadecimal base. The results shall be print on a screen and on a file named *ConvertionResults.txt*.<br>**All computation MUST be calculated using the basic algorithms, not functions or libraries.**<br><br>Req 3. The program shall include the mechanism to handle invalid data in the file. Errors should be displayed in the console and the execution must continue.<br><br>Req 4. The name of the program shall be convertNumbers.py<br><br>Req 5. The minimum format to invoke the program shall be as follows:<br>python convertNumbers.py fileWithData.txt<br><br>Req 6. The program shall manage files having from hundreds of items to thousands of items.<br><br>Req 7. The program should include at the end of the execution the time elapsed for the execution and calculus of the data. This number shall be included in the results file and on the screen.<br><br>Req 8. Be compliant with PEP8. | • Control structures<br>• Console Input output<br>• Error Handling | Record the execution.<br>Use files included in the assignment. |

TC1

First 10 values:

The binary value of 6980368.0 is: 11010101000001100010000

The Hex value of 6980368.0 is: 6.0A8.03.01.00.0

The binary value of 5517055.0 is: 10101000010111011111111

The Hex value of 5517055.0 is: 5.04.02.0EFF

The binary value of 1336159.0 is: 10100011000110101111

The Hex value of 1336159.0 is: 1.04.06.03.05.0F

The binary value of 6750185.0 is: 11001101111111111101001

The Hex value of 6750185.0 is: 6.06.0FFE9.0

The binary value of 1771937.0 is: 11011000010011010001

The Hex value of 1771937.0 is: 1.0B0.09.0A1.0

The binary value of 360952.0 is: 1011000000111111000

The Hex value of 360952.0 is: 5.08.01.0F8.0

The binary value of 5672561.0 is: 10101101000111001110001

The Hex value of 5672561.0 is: 5.06.08.0E7.01.0

The binary value of 916583.0 is: 11011111110001100111

The Hex value of 916583.0 is: DFC6.07.0

The binary value of 2700138.0 is: 1010010011001101101010

The Hex value of 2700138.0 is: 2.09.03.03.06.0A

The binary value of 9645053.0 is: 1001001100101011111111101

The Hex value of 9645053.0 is: 9.03.02.0BFD

**TC2**

First 10 values:

The binary value of 7116776.0 is: 1101100100101111111010000

The Hex value of 7116776.0 is: 6.0C9.07.0E8.0

The binary value of 1666340.0 is: 11001011010100100100

The Hex value of 1666340.0 is: 1.09.06.0D2.04.0

The binary value of 8886983.0 is: 10000111100110101011000111

The Hex value of 8886983.0 is: 8.07.09.0AC7.0

The binary value of 839365.0 is: 11001100111011000101

The Hex value of 839365.0 is: CCEC5.0

The binary value of 924280.0 is: 11100001101001111000

The Hex value of 924280.0 is: E1.0A7.08.0

The binary value of 1026310.0 is: 11111010100100000110

The Hex value of 1026310.0 is: FA9.00.06.0

The binary value of 1615293.0 is: 110001010010110111101

The Hex value of 1615293.0 is: 1.08.0A5.0BD

The binary value of 1063875.0 is: 100000011101111000011

The Hex value of 1063875.0 is: 1.00.03.0BC3.0

The binary value of 679035.0 is: 10100101110001111011

The Hex value of 679035.0 is: A5.0C7.0B

The binary value of 5201970.0 is: 100111101100000000110010

The Hex value of 5201970.0 is: 4.0F6.00.03.02.0

**TC3**

First 10 values:

he binary value of -39.0 is: -100111

The Hex value of -39.0 is: 2.07.0

The binary value of -36.0 is: -100100

The Hex value of -36.0 is: 2.04.0

The binary value of 8.0 is: 1000

The Hex value of 8.0 is: 8.0

The binary value of 34.0 is: 100010

The Hex value of 34.0 is: 2.02.0

The binary value of 17.0 is: 10001

The Hex value of 17.0 is: 1.01.0

The binary value of 49.0 is: 110001

The Hex value of 49.0 is: 3.01.0

The binary value of 5.0 is: 101

The Hex value of 5.0 is: 5.0

The binary value of 39.0 is: 100111

The Hex value of 39.0 is: 2.07.0

The binary value of 0.0 is: 0

The Hex value of 0.0 is: NA

The binary value of 33.0 is: 100001

The Hex value of 33.0 is: 2.01.0

TC4

First 10 values:

The binary value of -39.0 is: -100111

The Hex value of -39.0 is: 2.07.0

The binary value of -36.0 is: -100100

The Hex value of -36.0 is: 2.04.0

The binary value of 8.0 is: 1000

The Hex value of 8.0 is: 8.0

The binary value of 34.0 is: 100010

The Hex value of 34.0 is: 2.02.0

The binary value of 17.0 is: 10001

The Hex value of 17.0 is: 1.01.0

The binary value of 49.0 is: 110001

The Hex value of 49.0 is: 3.01.0

The binary value of 5.0 is: 101

The Hex value of 5.0 is: 5.0

The binary value of 0.0 is: 0

The Hex value of 0.0 is: NA

The binary value of 33.0 is: 100001

The Hex value of 33.0 is: 2.01.0

The binary value of 12.0 is: 1100

The Hex value of 12.0 is: C

| Programming Exercise | Description | Practice | Test Cases and Evidence |
|---|---|---|---|
| **3. Count Words** | Req1. The program shall be invoked from a command line. The program shall receive a file as parameter. The file will contain a words (presumable between spaces).<br><br>Req 2. The program shall identify all distinct words and the frequency of them (how many times the word "X" appears in the file). The results shall be print on a screen and on a file named *WordCountResults.txt*.<br>**All computation MUST be calculated using the basic algorithms, not functions or libraries.**<br><br>Req 3. The program shall include the mechanism to handle invalid data in the file. Errors should be displayed in the console and the execution must continue.<br><br>Req 4. The name of the program shall be wordCount.py<br><br>Req 5. The minimum format to invoke the program shall be as follows:<br>python wordCount.py fileWithData.txt<br><br>Req 6. The program shall manage files having from hundreds of items to thousands of items.<br><br>Req 7. The program should include at the end of the execution the time elapsed for the execution and calculus of the data. This number shall be included in the results file and on the screen.<br><br>Req 8. Be compliant with PEP8. | • Control structures<br>• Console Input output<br>• Error Handling<br>• String manipulation | Record the execution. Use files included in the assignment. |

**TC1**

**First 20 values:**

conservative - 2

mother - 1

tions - 1

pin - 1

sure - 1

regulatory - 1

shower - 1

uni - 1

dial - 1

photography - 1

buying - 1

firms - 1

nba - 1

father - 1

championship - 1

vagina - 1

fonts - 1

sparc - 1

explorer - 1

rl - 1

**TC2**

**First 20 values:**

holders - 4

amongst - 4

monaco - 4

filme - 4

doc - 4

kingston - 4

wood - 3

pre - 3

conduct - 1

kuwait - 1

literacy - 1

table - 1

parent - 1

olympic - 1

ht - 1

algebra - 1

ss - 1

norm - 1

females - 1

sq – 1

**TC3**

**First 20 values:**

notice - 3

flood - 2

pottery - 2

charity - 2

suggestion - 2

pairs - 2

blues - 2

pipe - 2

thumb - 2

reveals - 2

copy - 2

hurt - 2

neighbors - 1

manual - 1

political - 1

mozambique - 1

old - 1

holding - 1

fc - 1

ford – 1

**TC4**

**First 20 values:**

started - 3

literally - 2

ringtone - 2

za - 2

reached - 2

crazy - 2

javascript - 2

annual - 2

shown - 2

supplier - 2

physical - 2

data - 2

fought - 2

dramatically - 2

maiden - 2

contains - 2

panels - 2

racial - 2

charts - 2

navy - 2

**TC5**

**First 20 values:**

wilderness - 5

managed - 5

schools - 5

pets - 5

kg - 5

gps - 4

keeping - 4

travelling - 4

threats - 4

passion - 4

opens - 4

products - 4

webcams - 4

terrorist - 4

published - 4

petersburg - 4

manufactured - 4

suggestions - 4

margaret - 4

explain - 4

## Referencias:

Python enhancement proposals. PEP 8 – Style Guide for Python Code. (n.d.). https://peps.python.org/pep-0008/ Python Tutorial.