Eduardo García

OOP

14575

Team : 3
7. MATEO JAREN GARCIA GALARZA
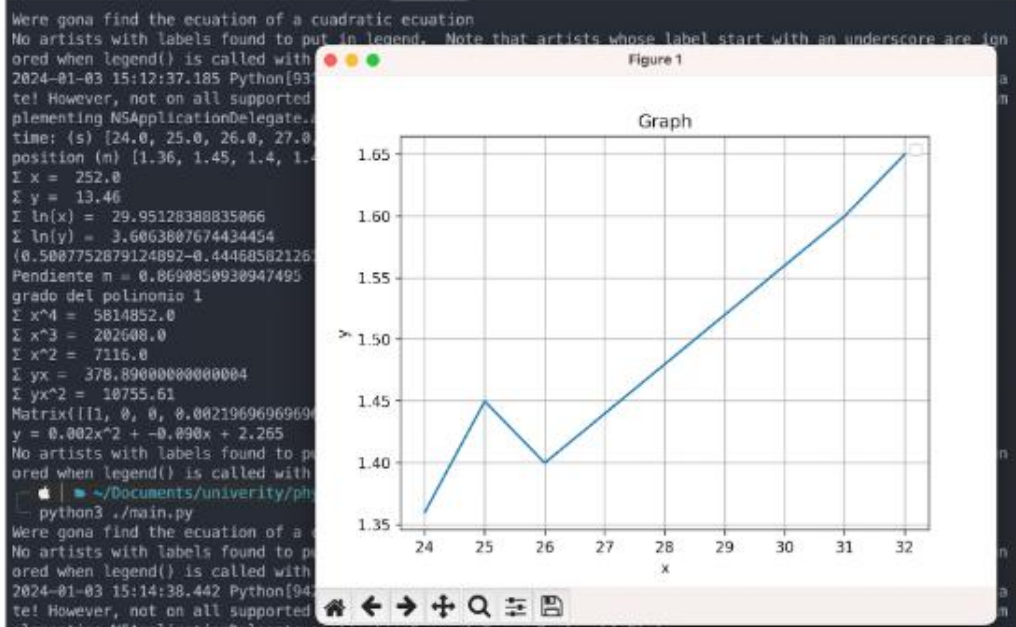8. EDUARDO DAVID GARCIA ROMERO
9. JUAN CARLOS GRANDA ARCOS

```python
import csv
import sympy as sp
import numpy as np
import matplotlib.pyplot as plt
import grapher

class EquationFinder:
    def __init__(self,file_path):
        print("Were gona find the ecuation of a cuadratic ecuation")
        with open(file_path, newline='', encoding='latin-1') as csvfile:
            reader = csv.reader(csvfile, delimiter=',')
            self.f1 = []
            self.f2 = []
            for row in reader:
                self.f1.append(row[0])
                self.f2.append(row[1])


    def depureColunn(self,columnData):
        columnData = [float(x.replace(',','.')) for x in columnData]
        return columnData

                                        (parameter) solvedSystem: Any
    def createFun(self,x,solvedSystem):
        return round(solvedSystem[0,3],3) * x**2 + round(solvedSystem[1,3],3) * x +round (solvedSystem[2,3],3)


    def ln(self, x):
```

```
time: (s) [24.0, 25.0, 26.0, 27.0
position (m) [1.36, 1.45, 1.4, 1.
Σ x =  252.0
Σ y =  13.46
Σ ln(x) =  29.95128388835066
Σ ln(y) =  3.6063807674434454
(0.5007752879124892,-0.444685821265
Pendiente m = 0.8690850930947495
grado del polinomio 1
Σ x^4 =  5814852.0
Σ x^3 =  202608.0
Σ x^2 =  7116.0
Σ yx =  378.89000000000004
Σ yx^2 = 10755.61
Matrix([[1, 0, 0, 0.0021969696969
y = 0.002x^2 + -0.090x + 2.265
```

```
    🍎 |  ► ~/Documents/univerity/phy
   python3 ./main.py
```

Figure 1

Graph


```python
import EcuationFinder


def main():
    ecuation_finder = EcuationFinder.EquationFinder("./tramo3.csv")
    ecuation_finder.findEcuation()


if __name__ == "__main__":
    main()
```


EquationFinder
-filePath
+depureColumn(columnData : List<float>) : List<float>
+createFun(x : List<float>) : Function
+computeLn(x : float) : float
+add(setX : List<float>) : float
+slope(y2 : float, x2 : float, y1 : float, x1 : float) : float
+findEcuation()

Grapher
-x : List<float>
-y : List<float>
+showGraph()
+graph()

Main

<<use>>
<<use>>
<<use>>