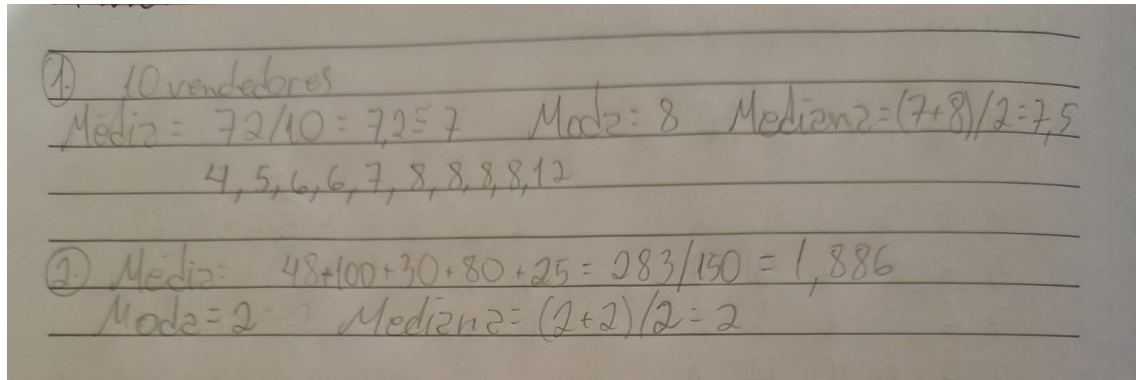


Aluno: Matheus da Silva Gomes

Ex.01, 02



Ex.03

```
"C:\Users\Matheus Gomes\AppData\Local\Programs\Python\Python310\python.exe" "C:\Users\Matheus Gomes\Desktop\Análise de Dados"
Média aritmética: 406.99
Moda: 145
Mediana: 365    Posição 35 de 70: 365    Posição 36 de 70: 365
Quartil 03: 568
Percentil ..P23: 154

Frequência
n_linhas = ou 7 ou 8 ou 9
Intervalo = de 87 em 87

De 120 até 207 = 17    24 %
De 207 até 294 = 15    21 %
De 294 até 381 = 6    9 %
De 381 até 468 = 8    11 %
De 468 até 555 = 4    6 %
De 555 até 642 = 6    9 %
De 642 até 729 = 5    7 %
De 729 até 816 = 5    7 %
De 816 até 903 = 4    6 %

Process finished with exit code 0
```

```
import statistics
# import numpy as np
# import sys
# import os
import exercicio_06 as ex

# sys.stdout.reconfigure(encoding='utf-8')

def verif_quartil(lista, n_quartil):
    if n_quartil == 1:
        position = int(len(lista) * 0.25)
        return lista[position]
    elif n_quartil == 2:
```

```

        position = int(len(lista) * 0.5)
        return lista[position]
    elif n_quartil == 3:
        position = int(len(lista) * 0.75)
        return lista[position]
    elif n_quartil == 4:
        return lista[-1]
    else:
        print("Valor quartil inválido!")
    return

def percentil(lista, n_percent):
    position = int(len(lista) * n_percent) - 1
    return lista[position]

numeros = "120 250 250 251 251 785 458 124 245 125 \
145 254 654 563 562 456 125 145 258 145 \
145 895 145 785 458 800 900 400 500 365 \
201 365 654 568 251 365 145 896 145 256 \
254 365 154 215 458 254 258 145 369 547 \
589 698 698 789 544 456 356 548 569 598 \
896 785 456 256 123 154 452 258 145 695 \
"

lista_tratada = list(map(int, numeros.split()))
media = statistics.mean(lista_tratada)
print(f"Média aritmética: {media:.2f}")

moda = statistics.mode(lista_tratada)
print(f"Moda: {moda}")

lista_ordenada = sorted(lista_tratada)
# print(lista_ordenada)
mediana = statistics.median(lista_ordenada)

print(f"Mediana: {int(mediana)}\t\tPosição 35 de 70: {lista_ordenada[35]}\
\t\tPosição 36 de 70: {lista_ordenada[36]}")

print(f"Quartil 0{3}: {verif_quartil(lista_ordenada, n_quartil=3)}")

print(f"Percentil ..P{23}: {percentil(lista_ordenada, n_percent=0.23)}")

at = ex.amplitude(lista_ordenada)
k = ex.qtde_linhas(len(lista_ordenada))
intervalo = ex.intervalo(at, k)
cont00 = min(lista_ordenada)
print("\nFrequência")
print(f'n_linhas = ou {k[0]} ou {k[1]} ou {k[2]}')
print(f'Intervalo = de {int(intervalo)} em {int(intervalo)}\n')
ex.cont_ocorrencias(lista_tratada, intervalo, cont00)

```

Ex.04

```
"C:\Users\Matheus Gomes\AppData\Local\Programs\Python\Python310\python.exe" "C:\Users\Matheus Gomes\l
Frequência:
n_linhas = ou 3 ou 4 ou 5
Intervalo = de 40 em 40
De 150 até 190 = 2    10 %
De 190 até 230 = 9    45 %
De 230 até 270 = 4    20 %
De 270 até 310 = 5    25 %
Média aritmética: 231.40
Moda: 210
Mediana: 215.5
Decil: 250
Quartil 2: 210
```

```
1      import calculos as calc
2
3      lista = "150 210 309 270 180 246 285 195 210 248 \
4      199 250 290 195 301 221 210 190 210 259"
5
6      calc.calc_frequency(lista)
7      calc.calc_media(lista)
8      calc.calc_moda(lista)
9      calc.calc_mediana(lista)
10     calc.decil(lista, 7)
11     calc.quartil(lista, 2)
12
```

```
import math
import statistics as stats
import sys

# sys.stdout.reconfigure(encoding='utf-8')

def cond_lista(lista):
    return list(sorted(map(int, lista.split()))))

def calc_moda(lista):
    lista = cond_lista(lista)
    moda = stats.mode(lista)
    print(f"Moda: {moda}")

def calc_media(lista):
    lista = cond_lista(lista)
    media = stats.mean(lista)
```

```

print(f"Média aritmética: {media:.2f}")

def calc_mediana(lista):
    lista = cond_lista(lista)
    print(f"Mediana: {stats.median(lista)}")

def amplitude(lista): # At
    xmax = max([x for x in lista])
    xmin = min([x for x in lista])
    return xmax - xmin

def qtde_linhas(n_elementos): # k
    opt0 = int(math.sqrt(n_elementos))
    opt1 = opt0 + 1
    opt2 = opt0 - 1
    return [opt2, opt0, opt1] # ex.: opt0 = 7 então 6, 7, 8

def intervalo(amplitude, qtde_linhas):
    n_real = [1, 2, 3, 4, 5, 6, 7, 8, 9]
    for i in range(len(n_real)):
        for j in range(len(qtde_linhas)):
            if (amplitude + n_real[i]) % qtde_linhas[j] == 0:
                ic = (amplitude + n_real[i]) / qtde_linhas[j]
                return ic
            else:
                continue
    return "erro"

def hit_count(lista, intervalo, cont):
    while cont < max(lista):
        n_ocorrencia = 0
        for i in range(len(lista)):
            if lista[i] >= cont and lista[i] < cont + intervalo:
                n_ocorrencia += 1
        print(f"De ", int(cont), " até", int(cont + intervalo), " = ", n_ocorrencia,
              "\t", int(round((n_ocorrencia/len(lista)) * 100)), "%")
        cont += intervalo

def verif_quartil(lista, n_quartil):
    lista = cond_lista(lista)
    if n_quartil == 1:
        position = int(len(lista) * 0.25)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 2:
        position = int(len(lista) * 0.5)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 3:
        position = int(len(lista) * 0.75)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 4:
        print(f"Quartil {n_quartil}: {lista[-1]}")

```

```

else:
    print("Valor quartil inválido!")
    return

def quintil(lista, n_quartil):
    lista = cond_lista(lista)
    if n_quartil == 1:
        position = int(len(lista) * 0.2)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 2:
        position = int(len(lista) * 0.4)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 3:
        position = int(len(lista) * 0.6)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 4:
        position = int(len(lista) * 0.8)
        print(f"Quartil {n_quartil}: {lista[position]}")
    elif n_quartil == 5:
        print(f"Quartil {n_quartil}: {lista[-1]}")
    else:
        print("Valor quartil inválido!")
    return

def decil(lista, n_decil):
    lista = cond_lista(lista)
    position = (int(len(lista) * (n_decil * 0.1)) - 1)
    print(f"Decil: {lista[position]}")

def percentil(lista, n_percent):
    lista = cond_lista(lista)
    position = int(len(lista) * n_percent) - 1
    print(lista[position])

def calc_frequency(lista):
    lista_tratada = cond_lista(lista)
    at = amplitude(lista_tratada)
    k = qtde_linhas(len(lista_tratada))
    interval = intervalo(at, k)
    cont00 = min(lista_tratada)
    print(f'Frequência:\nn_linhas = ou {k[0]} ou {k[1]} ou {k[2]}')
    print(f'Intervalo = de {int(interval)} em {int(interval)}')
    hit_count(lista_tratada, interval, cont00)

# if __name__ == "__main__": ///oq que essa bagaça faz mesmo??

```

