

Revisão para AP1 – Estrutura de Dados

Vetores:

1. Crie um programa que calcule a média de valores em um vetor.

```
# Solicita ao usuário inserir o número de elementos desejados no
vetor
n = int(input('Quantos numeros deseja inserir ?: '))

# Inicializa um vetor vazio para armazenar os valores inseridos
pelo usuário
vetor = []

# Loop para receber os valores do usuário e adicioná-los ao
vetor
for i in range(n):
    valor = float(input('Digite o {}º valor: '.format(i + 1)))
# Solicita ao usuário inserir o valor
    vetor.append(valor) # Adiciona o valor ao vetor

# Inicializa a variável para armazenar a soma dos valores
soma = 0

# Calcula a soma de todos os valores no vetor
for valor in vetor:
    soma += valor

# Calcula a média dividindo a soma pelo número de elementos no
vetor
media = soma / len(vetor)

# Exibe a média dos valores no vetor
print('A media dos valores no vetor é: {}'.format(media))
```

2. Crie um programa que dado um vetor, devolva 2 outros vetores com o quadrado e o cubo dos valores do vetor inicial.

```
# Inicializa três vetores vazios para armazenar os números
inseridos, seus cubos e seus quadrados
vetor = [] # Armazena os números inseridos
cubo = [] # Armazena os cubos dos números inseridos
quadrado = [] # Armazena os quadrados dos números inseridos

# Solicita ao usuário inserir a quantidade de números que deseja
adicionar ao vetor
x = int(input('Quantos numeros deseja inserir no vetor ?: '))

# Loop para inserir os números no vetor e calcular seus cubos e
quadrados
for i in range(x):
    num = int(input('Insira o {}º valor: '.format(i + 1))) #
    Solicita ao usuário inserir o número
    vetor.append(num) # Adiciona o número ao vetor
```

```

    cubo.append(num ** 2)    # Calcula o cubo do número e
    adiciona ao vetor cubo
    quadrado.append(num ** 3) # Calcula o quadrado do número e
    adiciona ao vetor quadrado

# Exibe os números inseridos no vetor, seus cubos e seus
quadrados
print(f'Numeros inseridos no vetor: {vetor}')
print(f'Numeros ao cubo dos inseridos: {cubo}')
print(f'Numeros ao quadrado dos inseridos: {quadrado}')

```

3. Escreva um programa que encontre o maior e o menor valor em um vetor.

```

# Inicializa um vetor vazio para armazenar os números inseridos
pelo usuário
vetor = []

# Pede ao usuário para inserir a quantidade de números que
deseja adicionar ao vetor
x = int(input('Quantos numeros deseja inserir ?: '))

# Loop para receber os números digitados pelo usuário e
adicioná-los ao vetor
for i in range(x):
    num = int(input(f'Insira o {i+1}º valor: ')) # Solicita ao
usuário inserir o número
    vetor.append(num) # Adiciona o número ao vetor

# Inicializa as variáveis para armazenar o maior e o menor
número no vetor
maior = vetor[0] # Inicializa a variável 'maior' com o primeiro
elemento do vetor
menor = vetor[0] # Inicializa a variável 'menor' com o primeiro
elemento do vetor

# Verifica os valores no vetor para determinar o maior e o menor
número
for num in vetor:
    if num > maior: # Verifica se o número atual é maior que
'maior'
        maior = num # Se sim, atualiza o valor de 'maior'
    if num < menor: # Verifica se o número atual é menor que
'menor'
        menor = num # Se sim, atualiza o valor de 'menor'

# Exibe o maior e o menor número, além dos números digitados
pelo usuário
print(f'Maior numero: {maior}')
print(f'Menor numero: {menor}')
print(f'Numeros digitados: {vetor}')

```

4. Crie um programa que conte o número de ocorrências de um determinado elemento em um vetor.

```

# Inicializa um vetor vazio para armazenar os números inseridos
pelo usuário
vetor = []

```

```

# Define o elemento a ser contado no vetor
elemento = 2

# Solicita ao usuário inserir a quantidade de números desejada
x = int(input('Insira quantos números deseja inserir: '))

# Inicializa um contador para contar as ocorrências do elemento no vetor
cont = 0

# Loop para receber os números digitados pelo usuário e adicioná-los ao vetor
for i in range(x):
    num = int(input(f'Insira o {i + 1}º número de 1 a 10: ')) # Solicita ao usuário inserir o número
    vetor.append(num) # Adiciona o número ao vetor
    if num == elemento: # Verifica se o número inserido é igual ao elemento desejado
        cont += 1 # Se sim, incrementa o contador

# Exibe o resultado do número de ocorrências do elemento no vetor
print(f'O número {elemento} apareceu {cont} vezes.')

```

5. Implemente um algoritmo para encontrar a soma de todos os elementos pares em um vetor.

```

# Inicializa um vetor vazio para armazenar os números inseridos pelo usuário
vetor = []

# Pede ao usuário inserir a quantidade de números desejada
x = int(input('Quantas vezes deseja inserir número?: '))

# Inicializa a variável para armazenar a soma dos números pares
soma = 0

# Loop para receber os números digitados pelo usuário e adicioná-los ao vetor
for i in range(x):
    num = float(input(f'Insira o {i + 1}º valor: ')) # Solicita ao usuário inserir o número
    vetor.append(num) # Adiciona o número ao vetor

    # Verifica se o número inserido é par
    if num % 2 == 0:
        soma += num # Se sim, adiciona o número à soma

# Exibe o resultado da soma de todos os elementos pares no vetor
print(f'A soma de todos os elementos pares no vetor é: {soma}')

```

6. Implemente um algoritmo para encontrar o segundo maior valor em um vetor.

```

# Inicializa um vetor vazio para armazenar os números inseridos
pelo usuário
vetor = []

# Solicita ao usuário inserir a quantidade de números que deseja
adicionar ao vetor
x = int(input('Quantas vezes deseja inserir numeros: '))

# Loop para receber os números digitados pelo usuário e
adicioná-los ao vetor
for i in range(x):
    num = int(input(f'Insira o {i + 1}º numero: ')) # Solicita
ao usuário inserir o número
    vetor.append(num) # Adiciona o número ao vetor

# Inicializa as variáveis maior e segundo maior com o primeiro
elemento do vetor
maior = vetor[0]
seg_maior = vetor[0]

# Loop para encontrar o maior número no vetor
for i in range(len(vetor)):
    if vetor[i] > maior:
        maior = vetor[i] # Atualiza o maior número encontrado

# Loop para encontrar o segundo maior número no vetor
for i in range(len(vetor)):
    if vetor[i] > seg_maior and vetor[i] < maior:
        seg_maior = vetor[i] # Atualiza o segundo maior número
encontrado

# Exibe o segundo maior número encontrado no vetor
print(f'O segundo maior numero: {seg_maior}')

```

7. Escreva um código em Python que receba, via teclado, 3 notas de um aluno, insira as notas em um vetor e imprima no console a maior e menor nota.

```

# Inicializa uma lista vazia para armazenar as notas inseridas
pelo usuário
notas = []

# Loop para solicitar ao usuário inserir as notas
for i in range(3):
    num = float(input(f'Insira a {i + 1}º nota: ')) # Solicita
ao usuário inserir a nota
    notas.append(num) # Adiciona a nota à lista de notas

# Encontra a maior e a menor nota na lista 'notas'
maior = max(notas) # Encontra a maior nota na lista 'notas'
menor = min(notas) # Encontra a menor nota na lista 'notas'

# Exibe a maior e a menor nota encontradas, formatadas com duas
casas decimais
print(f'A maior nota: {maior:.2f}')
print(f'A menor nota: {menor:.2f}')

```

Matrizes:

1. Implemente um algoritmo para encontrar o maior elemento em uma matriz.

```
# Define a matriz com os valores fornecidos
matriz = [[1, 20, 3],
          [40, 25, 6]]

# Inicializa a variável 'maior' com o primeiro elemento da
matriz
maior = matriz[0][0]

# Loop para percorrer todos os elementos da matriz
for i in range(2):
    for j in range(3):
        # Verifica se o elemento atual é maior que o valor atual
de 'maior'
        if matriz[i][j] > maior:
            maior = matriz[i][j] # Atualiza o valor de 'maior'
se necessário

# Exibe o maior elemento encontrado na matriz
print(f'Maior: {maior}')
```

2. Implemente um algoritmo para somar duas matrizes.

```
# Define as matrizes originais
matriz = [[1, 20, 3],
          [40, 45, 6]]

matriz_2 = [[1, 20, 3],
            [4, 4, 9]]

# Inicializa uma matriz vazia para armazenar o resultado da soma
das duas matrizes
matriz_limp = [[0, 0, 0],
               [0, 0, 0]]

# Loop para percorrer todas as posições das matrizes
for i in range(2):
    for j in range(3):
        # Soma os elementos correspondentes das duas matrizes e
atribui o resultado à matriz resultante
        matriz_limp[i][j] = matriz[i][j] + matriz_2[i][j]

# Exibe a matriz resultante
print(f'Resultado: {matriz_limp}')
```

3. Crie um programa que calcule a média dos elementos em uma matriz.

```
# Define a matriz com os valores fornecidos
matriz = [[10, 9, 9],
          [7, 8, 9]]
```

```

# Inicializa a variável 'media' com o valor zero
media = 0

# Loop para percorrer todos os elementos da matriz
for i in range(2):
    for j in range(3):
        media += matriz[i][j] # Adiciona o valor do elemento
                                atual à variável 'media'

# Divide a soma de todos os elementos pelo total de elementos na
matriz para calcular a média
media = media / 6

# Exibe a média calculada, formatada com duas casas decimais
print(f'A media calculada é: {media:.2f}')

```

4. Crie um programa que crie uma matriz, com as notas de 40 alunos e devolva a média da turma, a maior e a menor nota.

```

# Inicializa uma lista vazia para armazenar as notas dos alunos
alunos = []

# Loop para inserir as notas dos 5 alunos
for i in range(5):
    nota = float(input('Insira a nota do aluno: ')) # Solicita
    ao usuário inserir a nota do aluno
    alunos.append(nota) # Adiciona a nota à lista de notas dos
    alunos

# Calcula a média da turma
media = sum(alunos) / len(alunos)

# Encontra a maior e a menor nota da turma
maior = max(alunos)
menor = min(alunos)

# Exibe a média da turma, a maior e a menor nota, formatadas com
duas casas decimais
print(f'Media da turma: {media:.2f}')
print(f'Maior nota: {maior:.2f}')
print(f'Menor nota: {menor:.2f}')

```