



**Disciplina:** Paradigmas de Linguagens de Programação em Python

**Docente:** Heleno Filho

**Aluno:** José Gabriel da Silva Batista

## **Lista de Exercícios de Algoritmo Repetição e Vetores – UFBA**

**1.**

```
numeros = []

for i in range(10):
    numero = int(input(f"Digite o {i+1}º número inteiro: "))
    numeros.append(numero)

print("Lista de números:", end=" ")
for numero in numeros:
    print(numero, end=" ")
```

**2.**

```
numeros = []

for i in range(10):
    numero = int(input(f"Digite o {i+1}º número inteiro: "))
    numeros.append(numero)
numero_verificar = int(input("Digite um número para verificar na lista: "))

if numero_verificar in numeros:
    print(f"O número {numero_verificar} está na lista.")
```

else:

```
print(f"O número {numero_verificar} não está na lista.")
```

**3.**

```
numeros = []  
positivos = []  
negativos = []
```

```
for i in range(10):  
    numero = float(input(f"Digite o {i+1}º número real: "))  
    numeros.append(numero)  
    if numero > 0:  
        positivos.append(numero)  
    elif numero < 0:  
        negativos.append(numero)  
media = sum(numeros) / len(numeros)  
  
maior = max(numeros)  
menor = min(numeros)  
  
print(f"Média dos elementos: {media:.2f}")  
print(f"Maior elemento: {maior:.2f}")  
print(f"Menor elemento: {menor:.2f}")  
print(f"Quantidade de elementos positivos: {len(positivos)}")  
print(f"Quantidade de elementos negativos: {len(negativos)}")
```

**4.**

```
reajustadas  
nomes = []  
idades = []  
indenizacoes_reajustadas = []  
  
while True:  
    nome = input("Digite o nome completo do paciente (ou 'sair' para encerrar): ")  
  
    if nome.lower() == 'sair':  
        break  
  
    idade = int(input("Digite a idade do paciente: "))  
    valor_base = float(input("Digite o valor base de indenização: "))  
  
    if idade <= 12:  
        reajuste = 0.30  
    elif 13 <= idade <= 49:  
        reajuste = 0.10  
    elif 50 <= idade <= 65:  
        reajuste = 0.15
```

```

else:
    reajuste = 0.35

indenizacao_reajustada = valor_base * (1 + reajuste)

nomes.append(nome)
idades.append(idade)
indenizacoes_reajustadas.append(indenizacao_reajustada)

print("Resultados:")
for i in range(len(nomes)):
    print(f"Nome: {nomes[i]}, Idade: {idades[i]}, Indenização Reajustada: R${indenizacoes_reajustadas[i]:.2f}")

```

## 5.

```

import math

def calcular_norma(vetor):
    norma = math.sqrt(sum(x ** 2 for x in vetor))
    return norma

def calcular_soma(vetor1, vetor2, vetor3):
    soma = [x + y + z for x, y, z in zip(vetor1, vetor2, vetor3)]
    return soma

N = int(input("Digite o tamanho dos vetores: "))

vetor1 = []
vetor2 = []
vetor3 = []

print("Digite os valores do vetor 1:")
for i in range(N):
    valor = float(input(f"Digite o valor {i+1}: "))
    vetor1.append(valor)

print("Digite os valores do vetor 2:")
for i in range(N):
    valor = float(input(f"Digite o valor {i+1}: "))
    vetor2.append(valor)

print("Digite os valores do vetor 3:")
for i in range(N):
    valor = float(input(f"Digite o valor {i+1}: "))
    vetor3.append(valor)

norma1 = calcular_norma(vetor1)
norma2 = calcular_norma(vetor2)

```

```

norma3 = calcular_norma(vetor3)

maior_norma = max(norma1, norma2, norma3)

soma_vetores = calcular_soma(vetor1, vetor2, vetor3)

print(f"Norma do vetor 1: {norma1:.2f}")
print(f"Norma do vetor 2: {norma2:.2f}")
print(f"Norma do vetor 3: {norma3:.2f}")

if norma1 == maior_norma:
    print("O vetor 1 tem a maior norma.")
elif norma2 == maior_norma:
    print("O vetor 2 tem a maior norma.")
else:
    print("O vetor 3 tem a maior norma.")

print(f"Vetor Soma dos três vetores: {soma_vetores}")

```

## 6.

```

clientes = []

while True:
    nome = input("Digite o nome completo do cliente (ou 'sair' para encerrar): ")

    if nome.lower() == 'sair':
        break

    rg = input("Digite o RG do cliente: ")
    cpf = input("Digite o CPF do cliente: ")
    telefone = input("Digite o telefone do cliente: ")

    cliente = [nome, rg, cpf, telefone]
    clientes.append(cliente)

print("Cadastro de Clientes:")
for cliente in clientes:
    print(f"Nome: {cliente[0]}\nRG: {cliente[1]}\nCPF: {cliente[2]}\nTelefone: {cliente[3]}\n")

```

**7.**

```
elementos = []
```

```
for i in range(10):
```

```
    elemento = int(input(f"Digite o {i+1}º elemento inteiro: "))
```

```
    elementos.append(elemento)
```

```
valores_pares = sum(1 for elemento in elementos if elemento % 2 == 0)
```

```
print(f"Quantidade de valores pares na lista: {valores_pares}")
```

**8.**

```
numeros_pares = []
```

```
numeros_impares = []
```

```
tamanho_maximo = 5
```

```
while True:
```

```
    valor = int(input("Digite um valor inteiro (ou 0 para encerrar a leitura): "))
```

```
    if valor == 0:
```

```
        break
```

```
    if valor % 2 == 0:
```

```
        numeros_pares.append(valor)
```

```
    else:
```

```
        numeros_impares.append(valor)
```

```
    if len(numeros_pares) == tamanho_maximo:
```

```
        print("Lista de números pares:", numeros_pares)
```

```
        numeros_pares = []
```

```
    if len(numeros_impares) == tamanho_maximo:
```

```
        print("Lista de números ímpares:", numeros_impares)
```

```
numeros_impares = []
```

```
if numeros_pares:
```

```
    print("Lista de números pares restante:", numeros_pares)
```

```
if numeros_impares:
```

```
    print("Lista de números ímpares restante:", numeros_impares)
```

**9.**

```
gabarito = input("Digite o gabarito da prova com 20 questões (exemplo:  
ABCDEABCDEABCDEABCDE): ")
```

```
gabarito = list(gabarito)
```

```
resultados_alunos = []
```

```
num_alunos = 50
```

```
acertos_necessarios = 6 * 2 # Cada questão correta vale 0,5 ponto
```

```
for aluno in range(1, num_alunos + 1):
```

```
    nome = input(f"Digite o nome do aluno {aluno}: ")
```

```
    respostas = input(f"Digite as respostas do aluno {aluno} (exemplo:  
ABCDEABCDEABCDEABCDE): ")
```

```
    respostas = list(respostas)
```

```
    num_acertos = sum(1 for g, r in zip(gabarito, respostas) if g == r)
```

```
    nota_final = num_acertos * 0.5
```

```
    if nota_final >= acertos_necessarios:
```

```

        resultado = "APROVADO"
    else:
        resultado = "REPROVADO"

    resultados_alunos.append((nome, num_acertos, resultado))

for nome, num_acertos, resultado in resultados_alunos:
    print(f"Aluno: {nome}, Número de Acertos: {num_acertos}, Resultado: {resultado}")

```

## 10.

```

gabarito = input("Digite o gabarito da prova com 20 questões (exemplo: ABCDEABCDEABCDEABCDE): ")

```

```

gabarito = list(gabarito)

```

```

resultados_alunos = []

```

```

num_alunos = 50

```

```

acertos_necessarios = 6 * 2 # Cada questão correta vale 0,5 ponto

```

```

for aluno in range(1, num_alunos + 1):
    nome = input(f"Digite o nome do aluno {aluno}: ")
    respostas = input(f"Digite as respostas do aluno {aluno} (exemplo: ABCDEABCDEABCDEABCDE): ")

```

```

    respostas = list(respostas)

```

```

    num_acertos = sum(1 for g, r in zip(gabarito, respostas) if g == r)

```

```

    nota_final = num_acertos * 0.5

```

```

    if nota_final >= acertos_necessarios:

```

```
        resultado = "APROVADO"
    else:
        resultado = "REPROVADO"

    resultados_alunos.append((nome, nota_final, resultado))

for nome, nota_final, resultado in resultados_alunos:
    print(f"Aluno: {nome}, Nota Final: {nota_final}, Resultado: {resultado}")
```

## 11.

```
numeros_reais = []

for i in range(15):
    valor = float(input(f"Digite o {i+1}º número real: "))
    numeros_reais.append(valor)

while True:
    codigo = int(input("Digite um código (1 para ordem direta, 2 para ordem inversa, 0 para encerrar): "))

    if codigo == 0:
        break
    elif codigo == 1:
        print("Lista na ordem direta:", numeros_reais)
    elif codigo == 2:
        print("Lista na ordem inversa:", numeros_reais[::-1])
    else:
        print("Código inválido. Use 0, 1 ou 2 para escolher a ação.")
```

## 12.

```
a = int(input("Digite o valor de 'a': "))
b = int(input("Digite o valor de 'b': "))

N = int(input("Digite o tamanho da lista: "))
```



```
lista = []
```

```
for i in range(N):
```

```
    elemento = int(input(f"Digite o {i+1}º elemento da lista: "))
```

```
    lista.append(elemento)
```

```
contagem = sum(1 for elemento in lista if a <= elemento <= b)
```

```
print(f"A quantidade de elementos no intervalo [{a}, {b}] é: {contagem}")
```