

CENTRO UNIVERSITÁRIO BELAS ARTES DE SÃO PAULO
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS - EAD

FELIPE SCHAITEI - 24112424

ALGORITMOS E PROGRAMAÇÃO - MÓDULO 04

SÃO PAULO

2024

Lista de Exercícios Atividade 2

1 Escreva um algoritmo que considere a entrada de 4 notas de alunos e exiba qual é a maior e a menor pontuação entre eles.

```
aluno1 = int(input("digite a nota: "))
aluno2 = int(input("digite a nota: "))
aluno3 = int(input("digite a nota: "))
aluno4 = int(input("digite a nota: "))

notas = [aluno1, aluno2, aluno3, aluno4]

def bubble_sort_crescente(notas):
    for i in range(len(notas)):
        for j in range(len(notas) - 1 - i):
            if notas[j] > notas[j + 1]:
                notas[j], notas[j + 1] = notas[j + 1], notas[j]

def bubble_sort_descrescente(notas):
    for i in range(len(notas)):
        for j in range(len(notas) - 1 - i):
            if notas[j] < notas[j + 1]:
                notas[j], notas[j + 1] = notas[j + 1], notas[j]

lista_menor_nota = notas.copy()
lista_maior_nota = notas.copy()

bubble_sort_crescente(lista_menor_nota)
bubble_sort_descrescente(lista_maior_nota)

print("a maior nota é ", lista_maior_nota)
print("a menor nota é ", lista_menor_nota)
```

2 Crie um programa que lê 2 notas de um aluno e um caracter representando o tipo de média a ser calculada ('a' - aritmética, 'p' - Ponderada). Caso o tipo da média seja 'a', calcule e escreva na saída o valor da média aritmética, caso seja 'p' o calcule e escreva o valor da média ponderada com os seguintes pesos para as notas: 3 e 7.

```
total = 0
```

```
for i in range (2):
```

```
    valor = float(input("Escreva a nota " + str(i+1) + ": "))
```

```
    total = total + valor
```

```
media = total/2
```

```
print("a media é ", media)
```

3 Escreva um algoritmo que obtenha 2 números do tipo real e um caracter representando a operação. Para obter o resultado utilize a operação representada pelo caracter e mostre o resultado.

```
def operação (numero1, numero2, operacao):
```

```
    if operacao == "+":
```

```
        return numero1 + numero2
```

```
    else:
```

```
        operacao == "-" return numero1 - numero2
```

```
numero1 = 5
```

```
numero2 = 3
```

```
print(operacao)
```

4 Implemente um método que retorne a multiplicação de dois valores double.

```
def multiplicacao_dois_valores (primeiro_valor, segundo_valor):  
    return primeiro_valor * segundo_valor  
  
primeiro_valor = 3  
segundo_valor = 9  
multiplicacao_double = multiplicacao_dois_valores (primeiro_valor, segundo_valor)  
print( multiplicacao_double)
```

5 Implemente um método que retorne o valor de um arremesso de um dado. Ele recebe o número de faces como parâmetro

```
def valor_arremesso (faces_dado):  
  
    faces_dado = (1,2,3,4,5,6)  
    arremesso = faces_dado  
    return arremesso
```

6 Implemente um método que retorne o resto da divisão entre dois números, sem utilizar o mod (%)

```
def resto_divisao (numero, divisor):  
    quoc = int (numero / divisor)  
    res = numero - quoc * divisor  
    return res
```

7 Implemente um método que retorne o resto da divisão entre dois números, sem utilizar o mod (%)

```
def resto_divisao (numero, divisor):  
    quoc = int (numero / divisor)  
    res = numero - quoc * divisor  
    return res
```

8 Um jogo de Tiro possui dez fases em cada fase o jogador pode ter matado um número de inimigos (0 a 100). Crie um programa que permite que o jogador entre com o número de inimigos que ele matou em cada fase e exiba sua média.

```
numero_fases = 10  
numero_inimigos = 100  
  
kills_fases [ ]  
media_kills = kills_fases/10  
  
print("A sua média de kills é ", media_kills)
```

9 Supondo que a população de um formigueiro A seja da ordem de 90.000 formigas com uma taxa anual de crescimento de 3% e que a população do formigueiro B seja, aproximadamente, de 200.000 formigas com uma taxa anual de crescimento de 1.7%, fazer um algoritmo que calcule e escreva o número de anos necessários para que a quantidade de formigas do formigueiro A ultrapasse ou iguale a do formigueiro B, mantidas essas taxas de crescimento.

```
formigueiro_a = 90.000  
formigueiro_b = 200.00  
anos = 0  
  
while formigueiro_a < formigueiro_b  
    anos += 1  
    formigueiro_a += 0.03  
    formigueiro_b += 0.017
```

```
print ("passaram ", anos, "anos")
print("formigueiro A tem: ", formigueiro_a)
print("formigueiro B tem ", formigueiro_b)
```

10 Desenvolva um método que dado um número inteiro, ele exiba um triangulo reto com a base formada por um número de asteriscos igual ao número informado. Exemplo; número = 8

```
*****
*****
*****
*****
****
***
**
*
```

```
for element in nova_lista:
    print(element)
```

```
rows = 5
caractere = '*'
for i in range(rows):
    for j in range(i):
        print(caractere, end = ' ')
```

```
print("")
```

```
for i in range(1, rows + 1):
    for j in range(1, i + 1):
        print(j, end = ' ')
```

```
print("")
```

```
def print_triangle_pattern(symbol, rows):
```

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

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

```
            print(symbol, end=' ')
```

```
    print("")
```

```
print_triangle_pattern('&', 8)
```

11 Desenvolva um método que dado um número inteiro ímpar, ele exiba um losango com a linha central formada por um número de asteriscos igual ao número informado.

Exemplo; número = 7

```
      *
     ***
    *****
   ********
  *****
 *****
  ***
   *
```

```
for element in nova_lista:
```

```
    print(element)
```

```
rows = 7
```

```
caractere = '*'
```

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

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

```
print(caractere, end = ' ')
```

```
print("")
```

```
for i in range(1, rows + 1):
```

```
    for j in range(1, i + 1):
```

```
        print(j, end = ' ')
```

```
print("")
```

```
def print_diamond_pattern(symbol, rows):
```

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

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

```
            print(symbol, end=' ')
```

```
print("")
```

```
print_diamond_pattern('&', 8)
```