

LISTAS DE VETORES E MATRIZES

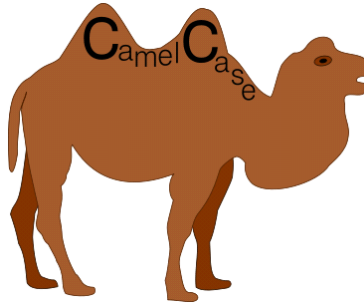
1. Faça um programa em Python que leia uma lista de 5 (cinco) números inteiros e mostre-o.
2. Faça um programa em Python que leia um lista de 10 (dez) númros inteiros e verifique quantos números digitados são pares e quantos são ímpares.
3. Escreva uma função em Python que receba uma lista de 10 (dez) números inteiros e devolva quantos dos números são primos.
4. O que será impresso pela função a seguir ?

```
def enigma():  
    M = [[0,0,0,0],[0,0,0,0],[0,0,0,0]]  
    for t in range(0,len(M[0])):  
        i = 0  
        while i<3:  
            M[i][t] = ((t*len(M))+i+1)*2-1  
            i = i+1  
        for t in range(0,len(M)):  
            print (M[t])
```

5. Escreva uma função que retorne uma matriz contendo a tabela verdade das funções **and** e **or**.
6. Escreva uma função em python que receba uma *string* e uma letra. Retorne **TRUE** se a letra estiver presente na *string*, caso contrário, retorne **FALSE**.
7. Crie uma função que verifique se duas listas, dadas com parâmetro, são iguais.
8. Escreva uma função em Python que receba uma lista com 20 valores inteiros, informados pelo usuário e apresente o maior, o menor e suas respectivas posições em que os mesmos foram informados. Caso existam números iguais mostre a posição da primeira ocorrência.
9. Escreva um programa em Python que leia dois vetores com 10 elementos cada e gere um terceiro vetor de 20 elementos, cujos valores deverão ser compostos pelos elementos intercalados dos dois outros vetores. Ao final o programa deverá mostrar os dois vetores originais e o terceiro vetor com os valores intercalados.
10. Dada uma frase, envie-a para uma função e retorne-a em CamelCase.

“**CamelCase** é a denominação em inglês para a prática de escrever palavras compostas ou frases, onde cada palavra é iniciada com Maiúscula e unidas sem espaços.”

Ex: “Minha casa, minha vida” → “MinhaCasa,MinhaVida”



Dica: Para transformar as letras minúsculas em maiúsculas use a tabela ASCII e as funções `ord()` e `chr()`.

CÓDIGO	SAÍDA
X= 'a' <code>print(X)</code>	a
Y= <code>ord('a')</code> <code>print(Y)</code>	97
Z= 65 <code>print(Z)</code>	65
W= <code>chr(65)</code> <code>print(W)</code>	A

11. Faça uma função em Python que leia um número inteiro e retorne um vetor contendo todos os divisores exatos deste número. Considere que serão fornecidos apenas números positivos como entrada.
12. Leia duas listas de inteiros e faça uma função que recebe os dois vetores e retorne uma terceira lista com a união entre os mesmos. O vetor união não deve conter números repetidos e deve ser ordenado de forma crescente. Os vetores não terão números repetidos em si.

ENTRADA	SAÍDA
Vet1 = [3,7,1,9,5,6,4] Vet2 = [1,7,5,8,2,0,3]	VetUniao =[0,1,2,3,4,5,6,7,8,9]

13. Faça um programa que calcule a média, moda e mediana de 100 (cem) lançamentos de dois dados, cada valor é representado pela soma da face superior dos dados e deve ser guardado em uma lista.

Para gerar os 100 (cem) números aleatórios, use a função `randint()` do módulo (ou biblioteca) `random`.

```

from random import randint      //importa a função randint() do módulo random

numero = randint(1,6)           //a variável numero recebe um valor inteiro entre 1 e 6

print(numero)                   //imprime o valor contido na variável numero

```

14. Faça um programa que leia uma matriz quadrada e verifique se ela é um quadrado mágico. Um quadrado mágico ocorre quando a soma de cada linha, cada coluna e das diagonais é igual.

				15
			↗	
2	7	6	→	15
9	5	1	→	15
4	3	8	→	15
↓	↓	↓	↘	
15	15	15		15

15. Faça um programa que leia duas matrizes, envie as matrizes para uma função que verifica se as mesmas podem ser multiplicadas. Caso positivo, envie as matrizes para uma outra função que efetue a multiplicação entre as matrizes e retornará uma terceira matriz, caso negativo, escreva na tela uma mensagem.

16. Crie uma função para calcular e imprimir os n primeiros números de Tribonacci.

“A série de Tribonacci consiste em: 1, 1, 2, 4, 7, 13, 24, 44, 81, 149, 274, 504,... Para calculá-la o primeiro elemento vale 1, o segundo elemento vale 1, o terceiro elemento vale 2, e daí por diante. Assim, o i-ésimo elemento vale o (i-1)-ésimo elemento somado ao (i-2)-ésimo elemento somado ao (i-3)-ésimo elemento. Exemplo: 13=7+4+2. Observe que n deve ser positivo.”

17. Crie uma função que verifique a interseção entre os números das Sequência de Fibonacci e Tribonacci. Utilize os números de 0 a 100.

Faça uma função retornar os números da Sequência de Fibonacci, uma para a Sequência de Tribonacci e outra para verificar a interseção entre as duas listas.

18. Crie uma função para determine o tamanho t da maior sequência de números iguais em uma lista A. Exemplo: Supor que sejam armazenados os seguintes valores para a lista A: [1,1,6,6,7,7,7,7,1,1,1], então t=4.