

Formativa 05 - Programação vetorial

```
import numpy as np
```

Matriz de Durer (quadrado mágico)

Matriz quadrada de dimensão 2 onde a soma de todas as linhas e soma de todas as colunas dão o mesmo valor. A soma da diagonal principal também dá o mesmo valor. Podemos chamar a função sum como um método do objeto array. A função np.diag extrai a diagonal principal de uma matriz.

```
D = np.array([[16,3,2,13],[5,10,11,8],[9,6,7,12],[4,15,14,1]],
dtype=np.float64)
print('--- D.sum(axis=0) soma as colunas -----')
print(D.sum(axis=0))
print('--- D.sum(1) soma as linhas -----')
print(np.sum(D,axis=1))
print('\n--- np.sum(D.diag()) soma a diagonal ----')
print(np.sum(np.diag(D)))

--- D.sum(axis=0) soma as colunas -----
[34. 34. 34. 34.]
--- D.sum(1) soma as linhas -----
[34. 34. 34. 34.]

---- np.sum(D.diag()) soma a diagonal ----
34.0
```

Quadrado mágico de dimensão ímpar

A função magic cria um quadrado mágico de dimensão ímpar.

```
def magic(N):
    # Cria um quadrado magico N x N magic square
    # N deve ser impar
    # https://scipython.com/book/chapter-6-numpy/examples/creating-a-magic-square/

    quadrado_magico = np.zeros((N,N), dtype=int)

    if N%2==0:
        return quadrado_magico

    n = 1
    i, j = 0, N//2 # Iniciar no meio da linha superior
```

```

while n <= N**2:      # Quando n = N ao quadrado terminou
    quadrado_magico[i, j] = n    # Salvar n na posição i,j
    n += 1
    # Mover uma posição para cima e uma para direita
    # Se ultrapassar o limite à direita ir para primeira coluna
    # Se ultrapassar o limite acima ir para última linha
    newi, newj = (i-1) % N, (j+1)% N
    if quadrado_magico[newi, newj]: # se a posição já estiver
preenchida
        i += 1                    # ir uma coluna para direita
    else:
        i, j = newi, newj

return quadrado_magico

```

Observar as propriedades da matriz criada pela função magic (quadrado mágico)

A soma de cada linha deve ser igual a soma de cada coluna e igual a soma da diagonal principal.

```

M = magic(5)
print('---- M----')
print(M)

print('\n---- M.sum(0) ----')
print(M.sum(0))

print('\n---- M.sum(1) ----')
print(M.sum(1))

print('\n---- np.sum(M.diag()) ----')
print(np.sum(np.diag(M)))

---- M----
[[17 24  1  8 15]
 [23  5  7 14 16]
 [ 4  6 13 20 22]
 [10 12 19 21  3]
 [11 18 25  2  9]]

---- M.sum(0) ----
[65 65 65 65 65]

---- M.sum(1) ----
[65 65 65 65 65]

---- np.sum(M.diag()) ----
65

```

Função para testar se uma matriz tem as propriedades do quadrado mágico

Utilizar o algoritmo seguinte para criar uma função que testa se uma matriz tem as propriedades do quadrado mágico:

- verificar se a matriz é quadrada: shape[0] tem que ser igual a shape[1]
- somar as colunas e colocar no array x0: utilizar a função sum (cada elemento de x0 é a soma de cada coluna)
- somar as linhas e colocar no array x1: utilizar a função sum (os elementos de x1 são a soma de cada linha)
- somar a diagonal da matriz m e colocar no array x2 (x2 = [np.sum(np.diag(m))])
- concatenar x0, x1 e x2 no array x (x=np.concatenate([x0, x1, x2]))
- se a matriz é um quadrado mágico, todos os elementos de x devem ser iguais
- encontrar os elementos únicos no array x: y = np.unique(x)
- testar se a quantidade de elementos únicos é igual a 1 (todos os elementos iguais): usar len(y)

```
def teste_magic(m):  
    # Colocar seu código aqui  
    if m.shape[0] != m.shape[1]:  
        return False  
    x0 = m.sum(0)  
    x1 = m.sum(1)  
    x2 = np.sum(np.diag(m))  
    x = np.concatenate([x0, x1, [x2]])  
    y = np.unique(x)  
    if len(y) != 1:  
        return False  
    return True
```

Usar a função teste_magic para testar as matrizes A, B, C, M3, M4, M5.

```
A = np.random.random((2,2))  
B = np.ones((2,2))  
C = np.zeros((2,2))  
M3 = magic(3)  
M4 = np.array([[16,3,2,13],[5,10,11,8],[9,6,7,12],[4,15,14,1]],  
dtype=np.float64)  
M5 = magic(5)  
  
print(teste_magic(A))  
print(teste_magic(B))  
print(teste_magic(C))  
print(teste_magic(M3))  
print(teste_magic(M5))
```

```
False  
True
```

```
True
True
True
```

Entrega

Programar a função teste_magic. Exportar para pdf. Fazer upload no AVA.