

## MAC0115 Introdução à Computação - IF

### Exercício-Programa 6 (EP6)

Entregar até 17 de dezembro de 2021

Escreva um programa, na linguagem `Python 3.x`, para realizar algumas transformações em matrizes. Uma das aplicações muito interessantes dessas transformações é em processamento de imagens que são representadas por matrizes de inteiros não negativos correspondentes a tonalidades de cinza.

#### Descrição das transformações

O seu programa deve implementar as transformações descritas a seguir. Cada uma dessas transformações deverá criar uma nova matriz, mantendo intacta a matriz inicial.

- Rebater uma matriz na vertical.
- Rotacionar uma matriz 90 graus no sentido horário (cuidado com matrizes retangulares).
- Extrair uma submatriz de uma matriz; ou seja, considerar a submatriz definida por uma área retangular delimitada pelos índices (linha e coluna) de duas posições da matriz: a primeira posição é a do canto superior esquerdo e a segunda é a do canto inferior direito da submatriz desejada.
- Obter a matriz da média de vizinhos. Cada elemento da nova matriz é a média aritmética inteira dos valores de quatro vizinhos (dois na horizontal e dois na vertical) do elemento correspondente na matriz inicial. Obs.: Os elementos na primeira linha, na última linha, na primeira coluna e na última coluna da nova matriz devem ser iguais aos da matriz inicial.
- Construir uma nova matriz somando a cada elemento da matriz inicial o valor da mediana de seus elementos. Além disso, essa nova matriz deve ter uma moldura com o valor da mediana.

#### Sobre a implementação e as funções que devem ser definidas:

As informações de uma matriz de inteiros não negativos, que será usada em seu programa, deverão estar num arquivo texto, sendo que cada linha do arquivo armazena os elementos da linha correspondente da matriz.

Seu programa deve executar uma sequência de operações, incluindo as transformações descritas anteriormente. Cada operação deverá ser escolhida pelo usuário, digitando a 'string' que nomeia a operação desejada.

A seguir listamos as possíveis transformações com os respectivos nomes.

- 'rebateV' - rebater uma matriz na vertical
- 'rotaciona' - rotacionar uma matriz 90 graus no sentido horário
- 'submatriz' - extrair uma submatriz de uma matriz
- 'mediaViz' - obter a matriz da média de vizinhos

- 'somaMed' - construir a matriz somando a mediana

Além das transformações, o usuário pode solicitar duas outras operações:

- 'listar' - listar todas as possíveis operações com os respectivos nomes
- 'finalizar' - finalizar a interação com o usuário

Nos casos em que a operação desejada pelo usuário é uma das transformações, o seu programa deve ler de um arquivo texto as informações de uma matriz, criar a matriz com a transformação desejada e imprimir todas as informações da matriz resultante.

No caso da transformação de extrair uma submatriz de uma matriz, o seu programa deve solicitar ao usuário que digite os índices (linha e coluna) de duas posições da matriz: a primeira posição é a do canto superior esquerdo e a segunda posição é a do canto inferior direito da submatriz desejada. O seu programa deve verificar se esses índices são válidos para a existência dessa submatriz. Se não forem válidos, o programa deve imprimir uma mensagem adequada.

No caso da transformação de obter a matriz da média de vizinhos, o seu programa deve verificar se a matriz lida tem pelo menos 3 linhas e pelo menos 3 colunas. Se não for o caso, o programa deve imprimir uma mensagem adequada.

A mediana de uma sequência de números pode ser obtida da seguinte forma:

- Colocar os números da sequência em ordem crescente ou decrescente.
- Se a quantidade de números da sequência é ímpar, a mediana é o valor central da sequência.
- Se a quantidade de números da sequência é par, a mediana é a média aritmética dos valores centrais da sequência. (Obs.: Neste ep vamos considerar média aritmética inteira.)

Implemente em seu programa, obrigatoriamente, todas as funções cujos protótipos estão descritos a seguir, sem nenhuma alteração, e sem alterar a ordem de definição das funções. Não utilize em seu programa nenhuma outra função além dessas obrigatórias.

```
def main():
    """ ( ) --> NoneType

    ... complete ...
    """

def cria_matriz(nlinhas, ncolunas, valor):
    """ (int, int, tipo do valor) --> matriz

    Cria uma matriz com nlinhas linhas e ncolunas colunas com todos os
    elementos iguais a valor. Retorna a matriz criada.
    """
```

```

def le_arq_cria_matriz():
    """ ( ) --> matriz

    Lê o nome de um arquivo texto contendo as informações de uma matriz
    (ou seja, cada linha do arquivo contém os elementos da linha
    correspondente da matriz).
    A função abre esse arquivo e ao mesmo tempo que lê os elementos da
    matriz, vai criando a matriz com os números lidos; fecha o arquivo
    e retorna a matriz criada.
    Obs.: Os elementos da matriz são números inteiros não negativos.
    """

def imprime_matriz(matriz):
    """ (matriz) --> NoneType

    Recebe uma matriz de inteiros e imprime-a no formato bidimensional
    de matriz e ajustada nas colunas.
    """

def rebate_vertical(a):
    """ (matriz) --> matriz

    Recebe uma matriz de inteiros a e constrói a matriz b
    resultante da aplicação da transformação correspondente em a.
    Retorna a matriz b.
    """

def rotaciona(a):
    """ (matriz) --> matriz

    Recebe uma matriz de inteiros a e constrói a matriz b
    resultante da aplicação da transformação correspondente em a.
    Retorna a matriz b.
    """

def submatriz(a, linSup, colSup, linInf, colInf):
    """ (matriz, int, int, int, int) --> matriz

    Recebe uma matriz de inteiros a e quatro inteiros:
    linSup e colSup que são os índices de linha e de coluna da posição
    no canto superior esquerdo da submatriz desejada;
    linInf e colInf que são os índices de linha e de coluna da posição
    no canto inferior direito da submatriz desejada.
    A função supõe que existe uma submatriz de a definida por esses
    dois índices; constrói a matriz b resultante da aplicação da
    transformação correspondente em a e retorna a matriz b.
    """

```

```

def media_vizinhos(a):
    """ (matriz) --> matriz

    Recebe uma matriz de inteiros a. A função supõe que a tem
    pelo menos 3 linhas e pelo menos 3 colunas.
    A função constrói a matriz b resultante da aplicação da
    transformação correspondente em a e retorna a matriz b.
    """

def mediana(seq):
    """ (list) --> int

    Recebe uma lista de inteiros seq.
    Determina e retorna a mediana de seq.
    """

def soma_mediana(a):
    """ (matriz) --> matriz

    Recebe uma matriz de inteiros a.
    A função constrói a matriz b resultante da aplicação da
    transformação correspondente em a e retorna a matriz b.
    """

def listar_operacoes():
    """ ( ) --> NoneType

    Escreve na tela os nomes e os significados das operações que um
    usuário pode escolher.
    """

```

## Exemplo de entrada e saída:

Veja a seguir um exemplo de entrada e saída (parcial) de uma execução deste programa.

. . . Escreva uma descrição sucinta deste programa (complete) . . .

-----

Digite o nome de uma operação: listar

Lista dos nomes e das respectivas operações:

```

rebateV      - rebater uma matriz na vertical
rotaciona    - rotacionar uma matriz 90 graus
submatriz    - extrair uma submatriz de uma matriz
mediaViz     - obter a matriz da média de vizinhos
somaMed      - construir a matriz somando a mediana
listar       - listar todas as operações com os seus nomes
finalizar    - finalizar a interação com o usuário

```

-----

Digite o nome de uma operação: rebateV

. . . leitura das informações de uma matriz (complete) . . .

Matriz com 7 linhas e 5 colunas:

9	4	5	0	8
1	3	10	3	2
1	7	5	7	9
1	6	3	15	2
2	0	3	8	10
1	0	0	1	16
13	12	7	1	2

Transformação realizada: rebater na vertical

Matriz resultante com 7 linhas e 5 colunas:

8	0	5	4	9
2	3	10	3	1
9	7	5	7	1
2	15	3	6	1
10	8	3	0	2
16	1	0	0	1
2	1	7	12	13

-----

Digite o nome de uma operação: rotaciona

. . . leitura das informações de uma matriz (complete) . . .  
. . . impressão das informações da matriz lida (complete) . . .

Transformação realizada: rotacionar 90 graus no sentido horário

Matriz resultante com 5 linhas e 7 colunas:

13	1	2	1	1	1	9
12	0	0	6	7	3	4
7	0	3	3	5	10	5
1	1	8	15	7	3	0
2	16	10	2	9	2	8

-----

Digite o nome de uma operação: submatriz

. . . leitura das informações de uma matriz (complete) . . .  
. . . impressão das informações da matriz lida (complete) . . .

Digite o índice de linha do canto superior esquerdo: 1

Digite o índice de coluna do canto superior esquerdo: 2

Digite o índice de linha do canto inferior direito: 4

Digite o índice de coluna do canto inferior direito: 3

Transformação realizada: extrair uma submatriz de (1,2) a (4,3)

Matriz resultante com 4 linhas e 2 colunas:

10	3
5	7
3	15
3	8

-----

Digite o nome de uma operação: mediaViz

. . . leitura das informações de uma matriz (complete) . . .  
. . . impressão das informações da matriz lida (complete) . . .

Transformação realizada: obter a matriz da média de vizinhos

Matriz resultante com 7 linhas e 5 colunas:

9	4	5	0	8
1	5	4	4	2
1	3	6	8	9
1	2	7	5	2
2	2	2	7	10
1	3	2	6	16
13	12	7	1	2

-----

Digite o nome de uma operação: somaMed

. . . leitura das informações de uma matriz (complete) . . .

Matriz com 4 linhas e 3 colunas:

9	4	5
1	3	10
1	7	2
12	8	10

Transformação realizada: construir a matriz somando a mediana

Matriz resultante com 6 linhas e 5 colunas:

6	6	6	6	6
6	15	10	11	6
6	7	9	16	6
6	7	13	8	6
6	18	14	16	6
6	6	6	6	6

-----

Digite o nome de uma operação: finalizar

-----

### Observações:

- O seu programa poderá usar somente os recursos da linguagem Python 3.x vistos em aula.
- Faça a entrada e a saída do seu programa de modo que as mensagens sejam como nos exemplos acima.
- O arquivo que você vai submeter, contendo o seu programa EP6, deverá ter o nome **EP6.py** .
- Leia atentamente as **Instruções para a entrega de EPs em Python**, e siga todos os passos e as recomendações descritas nesse documento para fazer a entrega (submissão) de programas no ambiente VPL do e-Disciplinas.