

## Introdução a Computação

### Atividade semanal 7

#### Variáveis compostas bidimensionais - Matrizes

1) Considere a matriz A dada por  $A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 2 & 1 \\ 2 & 2 & 1 \end{bmatrix}$

Execute o trecho de código manualmente. Quem será a matriz B? (mostre cada elemento)

```
for i in range(3):
    for j in range(3):
        B[i,j] = A[ A[i,j], A[j,i] ]
```

2) O trecho de programa a seguir deve receber uma matriz A e calcular o menor elemento de cada linha e de cada coluna de A. Ao fim da execução cada elemento de um vetor c[] deve armazenar o menor dos elementos da respectiva linha de A (ou seja, c[3] deve ser o elemento mínimo da linha 3 da matriz A). De forma similar, cada elemento de um vetor b[] deve armazenar o menor dos elementos da respectiva coluna de A (ou seja, b[2] deve ser o elemento mínimo da coluna 2). Supondo que A armazene valores aleatórios, como você pode modificar o trecho de programa a seguir para que ele realize esse procedimento de maneira correta? Explique seu raciocínio.

```
A = np.random.randint(0, 100, size=(M,N))
b = np.zeros(N)
c = np.zeros(M)
```

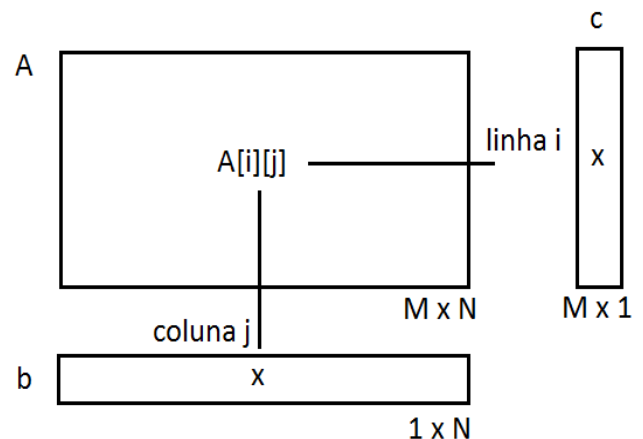
```
for i in range(M):
    c[i] = A[i,0]
```

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

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

```
        if A[i,j] < c[i]:
```

```
            if _____:
```



3. Dada uma matriz quadrada A n x n, com n definido pelo usuário, faça funções para:

- calcular a soma dos elementos da diagonal principal
- calcular a soma dos elementos da diagonal secundária
- determinar se a matriz é simétrica ou não
- encontrar a matriz triangular inferior
- calcular  $y = Ax$ , onde x é um vetor n x 1. Compare o resultado obtido pela sua implementação com o resultado da função dot do pacote numpy.
- calcular  $C = A B$ , onde B e C são matrizes n x n. Compare o resultado obtido pela sua implementação com o resultado da função dot do pacote numpy.