

Estrutura de Dados - Matrizes

Prof^a. Dr^a. Alana Morais
alanamm.prof@gmail.com

Objetivo

Entender as matrizes

Tipos de arrays

- Array (vetor)
- Array multidimensional (matriz)

Tipos de arrays

- ~~Array (vetor)~~
- Array multidimensional (matriz)

Array multidimensional (matriz)

- **Array convencional:**

`array1 = [1,2,3,4]`

1	2	3	4
---	---	---	---

Array multidimensional (matriz)

- **Array convencional:**

array1 = [1,2,3,4]

1	2	3	4
---	---	---	---

- **Lista multidimensional (matriz):**

1	2	3	4	5	6	7	8
---	---	---	---	---	---	---	---

array2 = [[1,2,3,4], [5,6,7,8]]

Array multidimensional (matriz)

- **Array convencional:**

array1 = [1,2,3,4]

1	2	3	4
---	---	---	---

- **Lista multidimensional (matriz):**

array2 = [
 [1,2,3,4] ,
 [5,6,7,8]
]

1	2	3	4
5	6	7	8

Operações com matrizes

- Criação de uma matriz;
- Varrendo valores da matriz;
- Remoção;
- Inserção.

Operações com matrizes

- **Criação de uma matriz;**
- Varrendo valores da matriz;
- Remoção;
- Inserção.

Criação de uma matriz

```
def main():  
    a = [ [1, 2, 3], [4, 5, 6] ]  
    print("primeira linha: ", a[0])  
    print("segunda linha: ", a[1])
```

```
main()
```

```
primeira linha: [1, 2, 3]  
segunda linha: [4, 5, 6]
```

Exercício

Crie um array que contenha na primeira posição o array ('ifpb', 'joão pessoa') e na segunda posição o array ('ufcg', 'campina grande').

Em seguida, imprima na tela.

Operações com matrizes

- Criação de uma matriz;
- **Varrendo valores da matriz;**
- Remoção;
- Inserção.

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
  
main()
```

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]
```

```
main()
```

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
        print('-----')  
  
main()
```

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

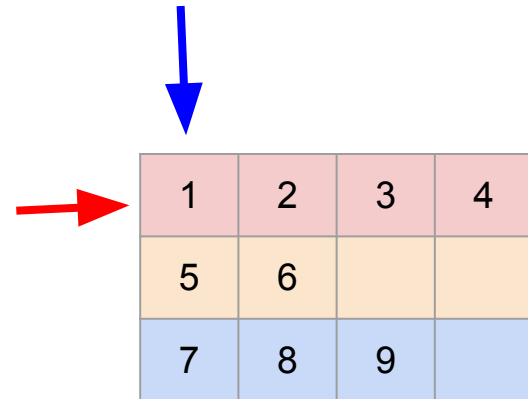
```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```



1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```

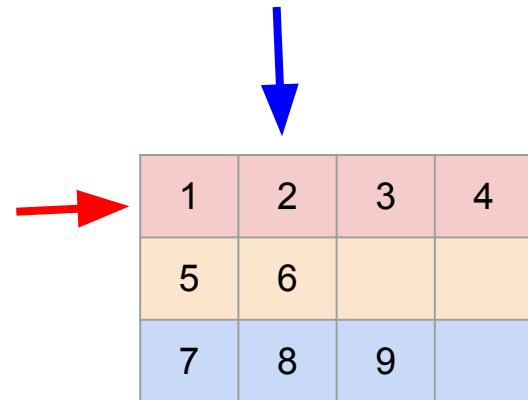


A 3x4 matrix is shown with colored cells. The first row (1, 2, 3, 4) is pink. The second row (5, 6, empty, empty) is orange. The third row (7, 8, 9, empty) is blue. A blue arrow points down to the first row, and a red arrow points right to the first column.

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```

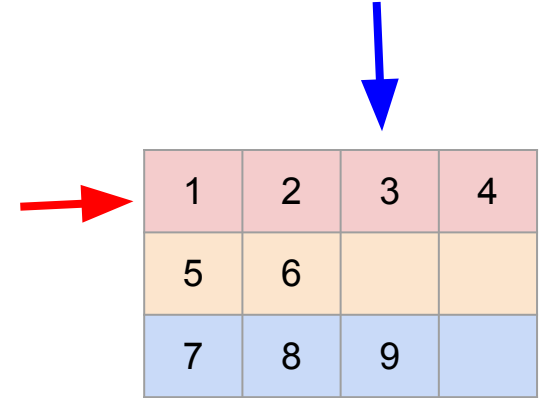


A 3x4 matrix is shown with colored cells. The first row (1, 2, 3, 4) is pink. The second row (5, 6, empty, empty) is orange. The third row (7, 8, 9, empty) is blue. A red arrow points to the first column, and a blue arrow points to the first row.

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```

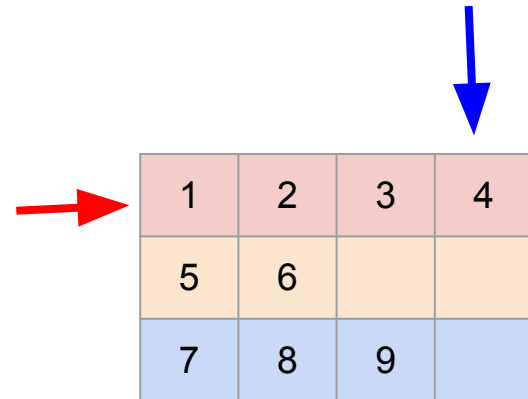


A 3x4 matrix is shown with colored cells. The first row (1, 2, 3, 4) is pink. The second row (5, 6, empty, empty) is orange. The third row (7, 8, 9, empty) is blue. A red arrow points to the first column, and a blue arrow points to the third column.

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```



A 3x4 matrix is shown with colored cells and arrows indicating iteration. A red arrow points to the first row, and a blue arrow points to the first column. The matrix contains the following values:

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

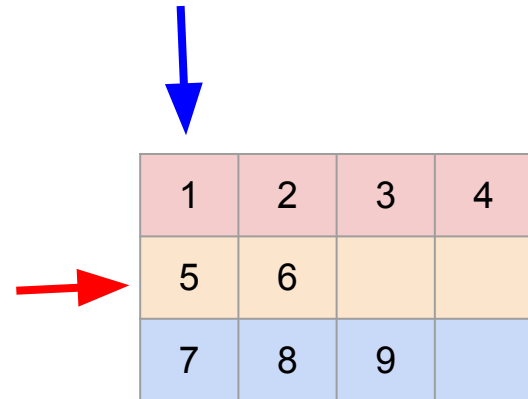
```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```



1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
        print('-----')  
  
main()
```

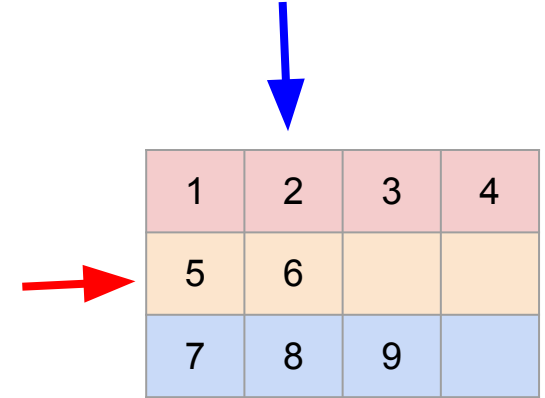


A 3x4 matrix is shown with colored cells. The first row (1, 2, 3, 4) is pink. The second row (5, 6, empty, empty) is orange. The third row (7, 8, 9, empty) is blue. A blue arrow points down to the first row, and a red arrow points right to the second row.

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
        print('-----')  
  
main()
```



A 3x4 matrix is shown with colored cells. The first row (1, 2, 3, 4) is pink. The second row (5, 6, empty, empty) is orange. The third row (7, 8, 9, empty) is blue. A blue arrow points down to the second row, and a red arrow points right to the second column.

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

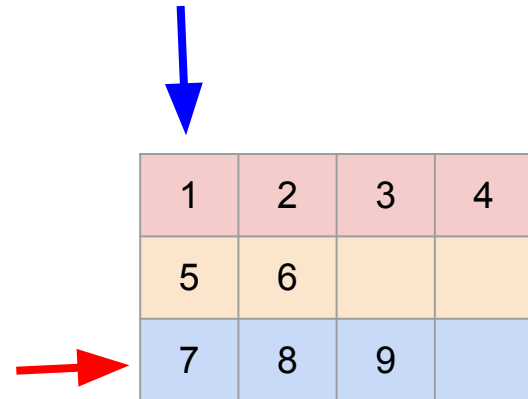
```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```



1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
        print('-----')  
  
main()
```

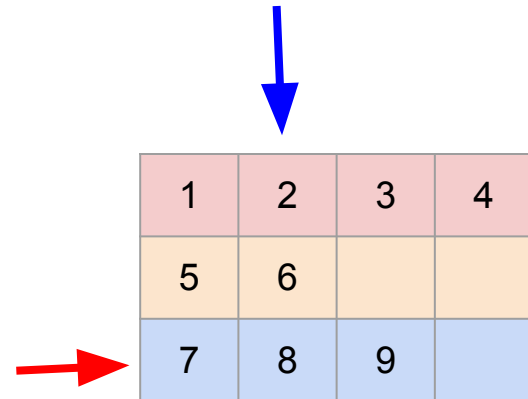


A 3x4 matrix is shown with colored cells and arrows. A blue arrow points down to the first row, and a red arrow points right to the first column. The cells contain the following values:

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
        print('-----')  
  
main()
```

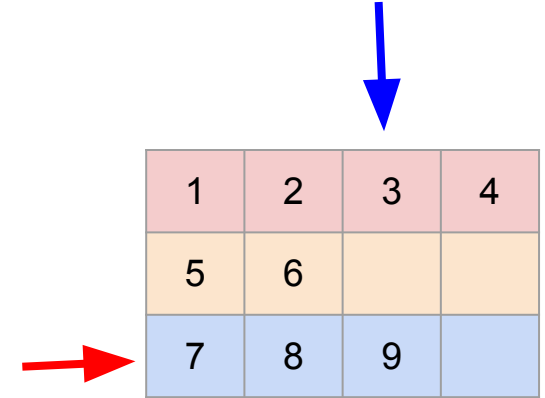


A 3x4 matrix is shown with cells colored in a 3x3 grid pattern. The first row has pink cells with values 1, 2, 3, 4. The second row has orange cells with values 5, 6, and two empty cells. The third row has blue cells with values 7, 8, 9, and one empty cell. A blue arrow points down to the cell containing 8, and a red arrow points right to the same cell.

1	2	3	4
5	6		
7	8	9	

Varrendo valores de uma matriz

```
def main():  
    a = [[1, 2, 3, 4], [5, 6], [7, 8, 9]]  
    for i in range(len(a)):  
        for j in range(len(a[i])):  
            print(a[i][j])  
            print('-----')  
  
main()
```



A 3x4 matrix is shown with colored cells and arrows. The first row (1, 2, 3, 4) is pink. The second row (5, 6, empty, empty) is orange. The third row (7, 8, 9, empty) is blue. A blue arrow points down to the first row, and a red arrow points right to the third row.

1	2	3	4
5	6		
7	8	9	

Exercício

Crie uma matriz que tenha o seguinte formato:

39	14	27
21	83	92
31	12	43

Em seguida, mostre todos os itens dessa matriz multiplicados por 7.

Operações com matrizes

- Criação de uma matriz;
- Varrendo valores da matriz;
- **Remoção de valores da matriz;**
- Inserção de valores da matriz.

Remoção de elementos da matriz

```
def main():  
    matriz = [ ["Bruna", "123131443"],  
               ["Alberto", "32141241"],  
               ["Filipe", "9474945"] ]
```

```
matriz[1].remove("Alberto")
```

```
matriz.pop(0)
```

```
main()
```

Remoção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]  
  
    for i in range(0, len(matriz)):  
        if "Alberto" in matriz[i]:  
            matriz[i].remove("Alberto")  
  
main()
```

Remoção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],    i = 0  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]  
  
    for i in range(0, len(matriz)):  
        if "Alberto" in matriz[i]:  
            matriz[i].remove("Alberto")  
  
main()
```


Remoção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"], i = 1  
        ["Filipe", "9474945"]  
    ]  
  
    for i in range(0, len(matriz)):  
        if "Alberto" in matriz[i]:  
            matriz[i].remove("Alberto")  
  
main()
```

Remoção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]    i = 2  
    ]  
  
    for i in range(0, len(matriz)):  
        if "Alberto" in matriz[i]:  
            matriz[i].remove("Alberto")  
  
main()
```

Remoção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"], i = 1  
        ["Filipe", "9474945"]  
    ]  
  
    for i in range(0, len(matriz)):  
        if "Alberto" in matriz[i]:  
            matriz[i].remove("Alberto")  
  
main()
```

Exercício

Crie uma matriz que tenha o seguinte formato:

39	14	27
21	83	92
31	12	43

Em seguida, remova os últimos itens de cada linha.

Operações com matrizes

- Criação de uma matriz;
- Varrendo valores da matriz;
- Remoção de valores da matriz;
- **Inserção de valores da matriz.**

Inserção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]  
  
    matriz.append("Eduardo")  
    matriz.append(["Thiago", "2342342334"])  
    matriz[0].append("10.0")  
main()
```

Inserção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]
```

```
matriz.append("Eduardo")
```

```
matriz.append(["Thiago", "2342342334"])
```

```
matriz[0].append("10.0")
```

```
main()
```

Bruna	123131443
Alberto	32141241
Filipe	9474945


Inserção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]
```

Bruna	123131443
Alberto	32141241
Filipe	9474945
Eduardo	

```
→ matriz.append("Eduardo")  
   matriz.append(["Thiago", "2342342334"])  
   matriz[0].append("10.0")  
main()
```


Inserção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]  
  
    matriz.append("Eduardo")  
     matriz.append(["Thiago", "2342342334"])  
    matriz[0].append("10.0")  
    main()
```

Bruna	123131443
Alberto	32141241
Filipe	9474945
Eduardo	
Thiago	2342342334

Inserção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]
```

```
    matriz.append("Eduardo")
```

```
    matriz.append(["Thiago", "2342342334"])
```

```
     matriz[0].append("10.0")
```

```
    main()
```

Bruna	123131443	10.0
Alberto	32141241	
Filipe	9474945	
Eduardo		
Thiago	2342342334	

Inserção de elementos da matriz

```
def main():  
    matriz = [  
        ["Bruna", "123131443"],  
        ["Alberto", "32141241"],  
        ["Filipe", "9474945"]  
    ]
```

```
    matriz.append("Eduardo")
```

```
    matriz.append(["Thiago", "2342342334"])
```

```
    matriz[0].append("10.0")
```

```
main()
```

0	Bruna	123131443	10.0
1	Alberto	32141241	
2	Filipe	9474945	
3	Eduardo		
4	Thiago	2342342334	

Exercício

Crie uma matriz que contenha na primeira linha o array [1,2,3,4] e na segunda linha o array [5,6,7,8].

Depois, insira um valor, digitado por você, na última posição de todas as linhas.

Exercício

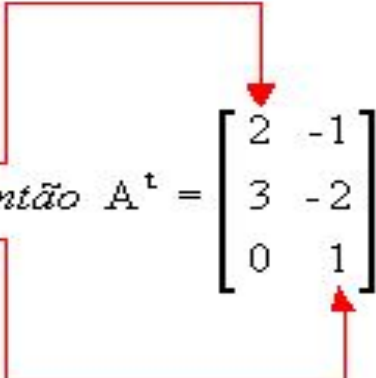
Leia uma matriz 4 x 4, conte e escreva quantos valores maiores que 10 ela possui.

Exercício

Crie uma matriz que armazene nome, matrícula e data de nascimento de 3 alunos.

Exercício

Criar um algoritmo que leia uma matriz $A_{N \times N}$ ($N \leq 10$) e calcule a respectiva matriz transposta A^t



Se $A = \begin{bmatrix} 2 & 3 & 0 \\ -1 & -2 & 1 \end{bmatrix}$, então $A^t = \begin{bmatrix} 2 & -1 \\ 3 & -2 \\ 0 & 1 \end{bmatrix}$

Atividade

Crie um tabuleiro de jogo da velha, usando uma matrizes de caracteres 3x3, onde o usuário pede o número da linha (1 até 3) e o da coluna (1 até 3).

A cada vez que o usuário entrar com esses dados, colocar um 'X' ou 'O' no local selecionado.

Dúvidas?



alanamm.prof@gmail.com