

Programação de Computadores III

Matriz

Bibliografia

- DOWNEY A. Pense em Python. São Paulo: Novatec, 2016.**
- MENEZES, N. N. C. Introdução à Programação com Python. São Paulo: Novatec, 2014.**
- WAZLAWICK, R. S. Introdução a Algoritmos e Programação com Python. 1. ed. Elsevier, 2017, 232p.**
- Sedgewick, Robert, Kevin Wayne, and Robert Dondero. Introduction to programming in Python: An interdisciplinary approach. Addison-Wesley Professional, 2015**
- MARTELLI A., ASCHER D. Python Cookbook. O'Reilly, 2002, 575 pages.**
- ASCHER D., LUTZ M. Aprendendo Python, Editora Bookman, 2ª edição, 2007, 566 páginas.**
- FORBELLONE, A. L. V.; EBERSPÄCHER, H. F.. Lógica de programação : a construção de algoritmos e estruturas de dados. 3.ed. São Paulo:Prentice Hall, 2005. xii, 218p.**
- BORGES, L. E. Python para Desenvolvedores. São Paulo: Novatec, 2014.**

Matriz

- As matrizes em Python podemos enxergar elas como lista dentro de outra lista.
- Como já estudamos em Prog II sobre variáveis compostas, e listas, podemos facilmente trabalhar com matrizes, pois iremos utilizar o conceito de lista, porém uma lista armazenando outra lista e assim criando uma matriz.
- Lembrando que todos os comandos utilizados em lista, podem ser utilizados em matrizes, porém com algumas adequações e cuidados.

Matriz

- O conceito de matriz matematicamente, matriz é toda estrutura com formato $n \times m$, ou seja, se colocar-mos uma lista dentro de outra lista estaremos fazendo exatamente isso.

$$M = [1 \ 2 \ 3 \ 4]$$

- Em python representamos essa matriz da seguinte forma:

Python

```
1. M = [[1, 2], [3, 4]]
```

-

Matriz

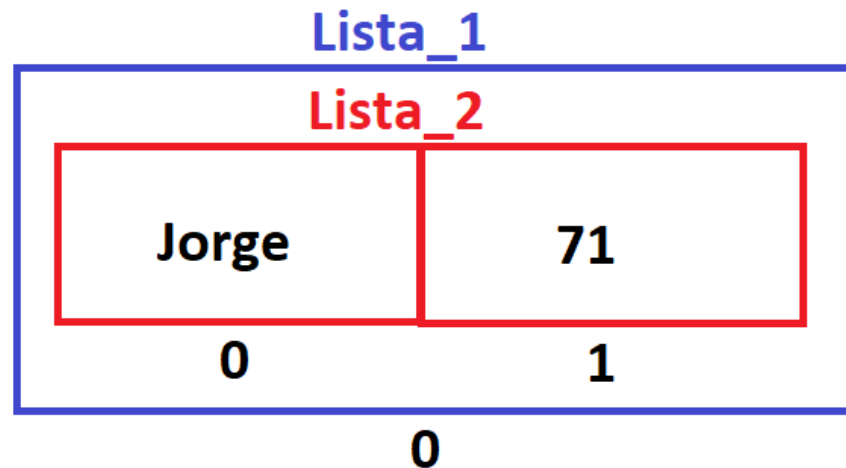
```
1  ...
2
3  AMBIENTE DE PROGRAMAÇÃO
4
5  ...
6  lista_1=[]
7  lista_2=["Thiago", 30]
8
9  lista_1.append(lista_2)
10 print(lista_1)
```

[['Thiago', 30]]

- Reparem no resultado, que temos uma lista dentro de outra lista.

Matriz

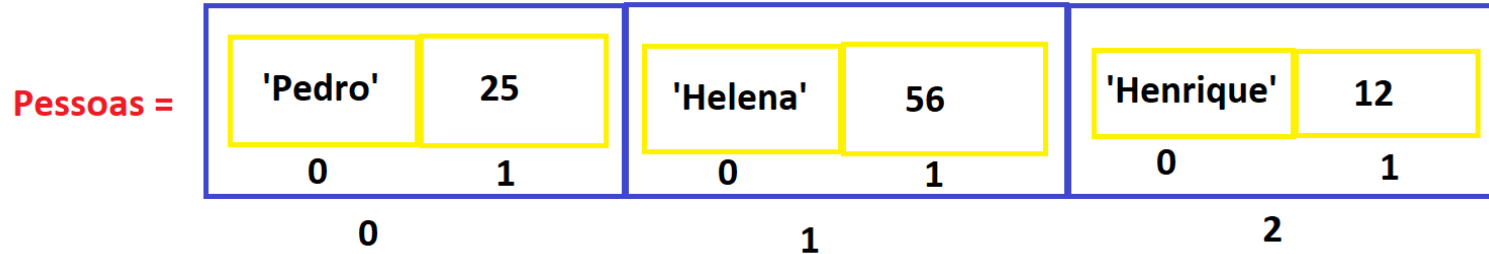
- Preste atenção na estrutura, note que em python uma matriz, é composta de uma lista contendo dentro dela outras listas que irão representar os eixos X e Y da matriz.



- `Lista_1=[Lista_2=["Jorge",71]]`
- Reparem que na lista_1 possui outra estrutura de lista dentro dela que nesse caso é a lista_2.

Matriz

- Neste exemplo temos uma lista chamada pessoas onde dentro dela contem outras três listas. Dessa maneira possuímos uma lista externa que tem três índices, e em cada índice encontramos outra lista que possui mais duas informações armazenadas em índices diferentes.



Pessoas = [['Pedro',25] , ['Helena',56] , ['Henrique',12]]

Matriz

```
1 '''
2
3 AMBIENTE DE PROGRAMAÇÃO
4
5 '''
6
7 pessoas=[["Fabricio",28],["Thiago",30],["Bruna",23],["Carla",27]]
8
9 print(pessoas[0][0])      Fabricio
10
11 print(pessoas[3][0])     Carla
12
13 print(pessoas[2][1])     23
```

- Quando queremos exibir um elemento específico basta colocar o número do índice correspondente a ele.

Exercício 01

- Crie um programa onde o usuário possa digitar sete valores numéricos e cadastre-os em uma lista única que mantenha separados os valores pares e ímpares. No final mostre os valores pares e ímpares em ordem crescente.

Exercício 01 - Resolvido

```
1 '''
2
3 AMBIENTE DE PROGRAMAÇÃO
4
5 '''
6
7 par = []
8 impar = []
9 geral = []
10
11 for i in range (7):
12     num = int(input('Forneça um numero '))
13     if num % 2 == 0:
14         par.append(num)
15     else:
16         impar.append(num)
17 par.sort()
18 impar.sort()
19 geral.append(par)
20 geral.append(impar)
21
22 print(geral)
```

Forneça um numero 3

Forneça um numero 2

Forneça um numero 5

Forneça um numero 6

Forneça um numero 4

Forneça um numero 5

Forneça um numero 3

[[2, 4, 6], [3, 3, 5, 5]]

Exercício 02

- Crie um programa que crie uma matriz 3X3 (dimensão) e preencha com valores lidos pelo teclado. No final mostre a matriz na tela, com a formatação correta.

Exercício 02 - Resolvido

```
1 '''
2
3 AMBIENTE DE PROGRAMAÇÃO
4
5 '''
6 linha=[]
7 coluna=[]
8
9 for l in range (3):
10     for c in range(3):
11
12         print('forneça o elemento (' ,l,c,')')
13         num=int(input())
14         coluna.append(num)
15
16
17     linha.append(coluna[:])
18     coluna.clear()
19
20
21
22 for l in linha:
23     for c in l:
24         print(c,end=' ' )
25     print()
26 print('*'*35)
```

forneça o elemento (0 0)

2
forneça o elemento (0 1)

4
forneça o elemento (0 2)

3
forneça o elemento (1 0)

2
forneça o elemento (1 1)

5
forneça o elemento (1 2)

6
forneça o elemento (2 0)

7
forneça o elemento (2 1)

5
forneça o elemento (2 2)

4
2 4 3
2 5 6
7 5 4
