

# **Aula 8 – Estrutura de Dados continuação**

# Listas - Fatias

Podemos usar a notação de fatias (slices) em listas.

```
>>> lista = ['a', 2, [3, 'f'], 'q']
```

```
>>> lista [1:]  
[2, [3, 'f'], 'q']
```

```
>>> lista [:1]  
['a']
```

[índice inicial : índice final+1]

```
>>> lista [1:2]  
[2]
```

```
>>> lista [0:-1]  
['a', 2, [3, 'f']]
```

# Listas - Fatias

**Atribuição:** ao atribuir uma *sequência* a uma fatia, os elementos desta devem ser substituídos pelos elementos daquela.

```
>>> lista = [1, 2, 3, 4, 5]
```

[1, 2, 3, 4, 5]

[1, "a", "b", 4, 5]

lista[1:3] = ["a", "b"]

[1, 2, 3, 4, 5]

[1, 2, "a", "b"]

lista[2:] = ["a", "b"]

# Listas - Fatias

**Atribuição:** ao atribuir uma *sequência* a uma fatia, os elementos desta devem ser substituídos pelos elementos daquela.

```
>>> lista = [1,2,3,4,5]
```

[1, 2, 3, 4, 5]

["a", "b", 3, 4, 5]

lista[:2] = ["a","b"]

[1, 2, 3, 4, 5]

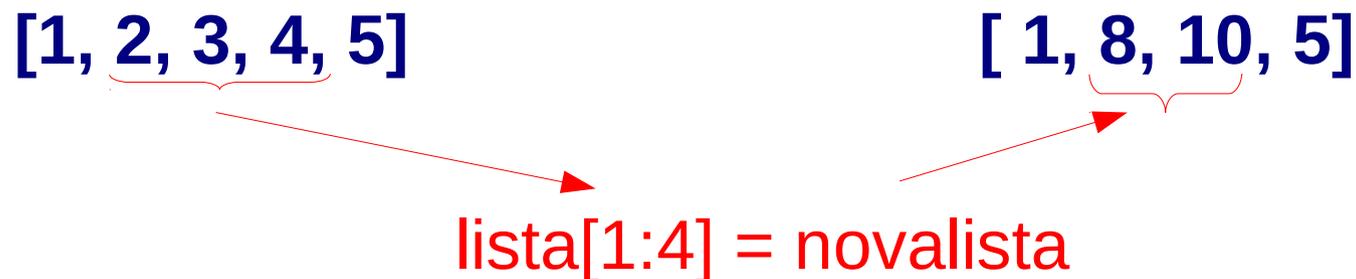
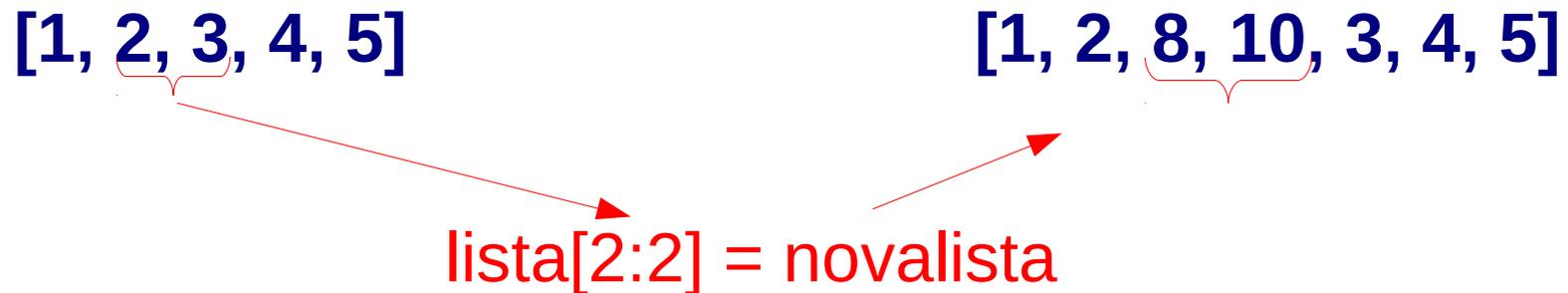
[ ["a", "b"], 3, 4, 5]

lista[:2] = [ ["a","b"] ]

# Listas - Fatias

**Atribuição:** ao atribuir uma *sequência* a uma fatia, os elementos desta devem ser substituídos pelos elementos daquela.

```
>>> lista = [1, 2, 3, 4, 5]
>>> novalista = [8, 10]
```



# Listas - Fatias

**Atribuição:** ao atribuir uma *sequência* a uma fatia, os elementos desta devem ser substituídos pelos elementos daquela.

## **Exercício**

```
>>> lista = [1, 2, 3, 4, 5]
```

```
>>> lista [1:1] = ['z']
```

```
>>> lista [1:3] = [[7]]
```

```
>>> lista [1:-1]= [8, 9, 10]
```

```
>>> lista[:3]="xyz"
```

```
>>> lista[:3]="a, b, c"
```

```
>>> lista[:2]=1, 2, 3
```

# Listas - Fatias

**Atribuição:** ao atribuir uma *sequência* a uma fatia, os elementos desta devem ser substituídos pelos elementos daquela.

## Exercício

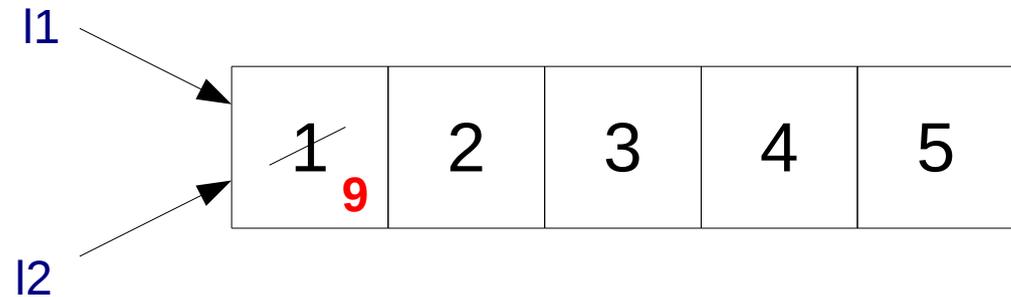
```
>>> lista = [1,2,3,4,5]
>>> lista [1:1] = ['z']
      [1, 'z', 2, 3, 4, 5, 6]
>>> lista [1:3] = [[7]]
      [1, [7], 3, 4, 5, 6]
>>> lista [1:-1]= [8,9,10]
      [1, 8, 9, 10, 6]
>>> lista[:3]="xyz"
      ['x', 'y', 'z', 10, 6]
>>> lista[:3]="a,b,c"
      ['a', ',', 'b', ',', 'c', 10, 6]
>>> lista[:2]=1,2,3
      [1, 2, 3, 'b', ',', 'c', 10, 6]
```

# Listas - Fatias

**Atribuição:** *cuidado quando fizer cópia de listas!*

## Exemplo

```
>>> l1 = [1, 2, 3, 4, 5]
>>> l2 = l1
>>> l1
[1, 2, 3, 4, 5]
>>> l2
[1, 2, 3, 4, 5]
>>> l2[0]=9
>>> l2
[9, 2, 3, 4, 5]
>>> l1
[9, 2, 3, 4, 5]
```



# Listas - Fatias

**Atribuição:** *cuidado quando fizer cópia de listas!*

## Exemplo

```
>>> l1 = [1, 2, 3, 4, 5]
```

```
>>> l2 = l1[:]
```

```
>>> l1  
[1, 2, 3, 4, 5]
```

```
>>> l2  
[1, 2, 3, 4, 5]
```

```
>>> l2[0]=9
```

```
>>> l2  
[9, 2, 3, 4, 5]
```

```
>>> l1  
[1, 2, 3, 4, 5]
```



# Listas - Fatias

**Incremento:** *podemos usar incremento / decremento para selecionar os elementos de uma lista.*

## Exemplo

```
>>> lista = [1,2,3,4,5,6]
```

```
>>> lista[0:-1:2]  
[1, 3, 5]
```

```
>>> lista[5:0:-1]  
[6, 5, 4, 3, 2]
```

```
>>> lista[0:-1:3]  
[1, 4]
```

```
>>> lista[::-1]  
[6, 5, 4, 3, 2, 1]
```

# Listas - Fatias

**Incremento:** *podemos usar incremento / decremento para selecionar os elementos de uma lista.*

**ATENÇÃO:** *Se um incremento de fatia é diferente de 1, uma atribuição à fatia deve ter o mesmo número de elementos.*

```
>>> l = [1,2,3,4,5]
```

```
>>> l[0::2] = ['x','y','z']
```

```
>>> l
['x',2, 'y', 4, 'z']
```

```
>>> l[0::2] = [6,7]
```

```
Traceback (most recent call last):
```

```
File "<pyshell#17>",line 1, in -toplevel-
```

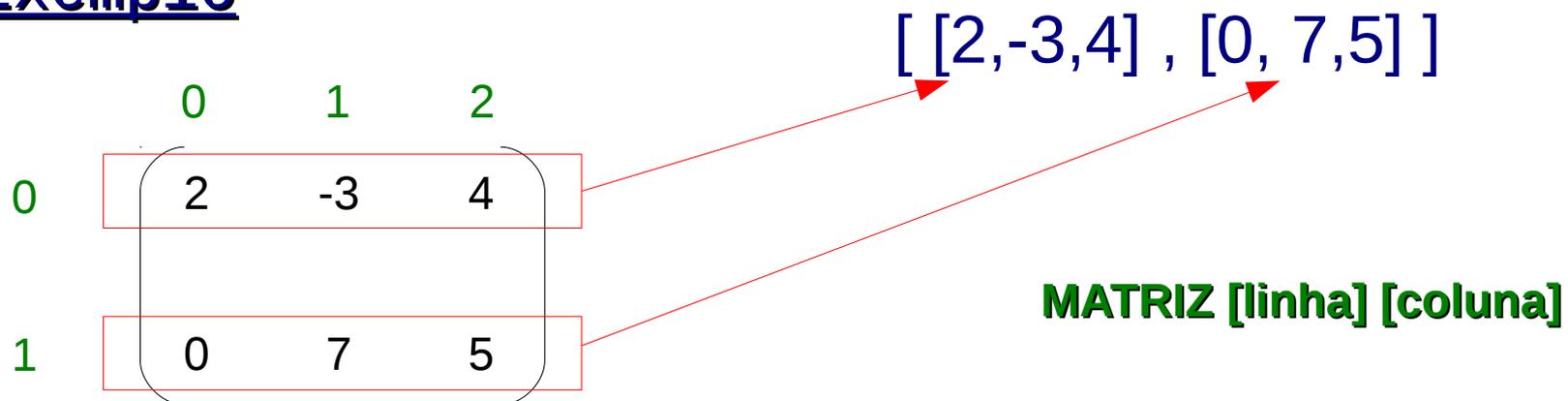
```
    l [0::2] = [6,7]
```

```
ValueError: attempt to assign sequence of size 2  
to extended slice of size 3
```

# Listas - Matrizes

Podemos usar listas para armazenar e manipular matrizes.

## Exemplo



```
MATRIZ = [ [2, -3, 4],[0, 7, 5] ]  
for i in range(2):  
    for j in range (3):  
        print MATRIZ[i][j]
```

**MATRIZ [0] [0] = 2**

**MATRIZ [1] [0] = 0**

**MATRIZ [0] [1] = -3**

**MATRIZ [1] [1] = 7**

**MATRIZ [0] [2] = 4**

**MATRIZ [1] [2] = 5**

# Listas - Matrizes

**Exercício:** Construa uma matriz 4x3 com valores iguais a zero. Imprima a matriz.

Prog. 1

```
matriz = []
linha = [0]*3
for i in range(4):
    matriz = matriz+[linha]
for i in range(4):
    for j in range (3):
        print matriz[i][j],
    print
```

Prog. 2

```
matriz = []
for i in range(4):
    matriz = matriz.append([0]*3)
for i in range(4):
    for j in range (3):
        print matriz[i][j],
    print
```

**ATENÇÃO:** estes programas são equivalentes?

# Listas - Matrizes

**Exercício:** Faça um programa que permita ao usuário construir uma matriz de inteiros  $N \times M$ . Imprima a matriz que foi fornecida.

# Listas - Matrizes

**Exercício:** Faça um programa que permita ao usuário construir uma matriz de inteiros  $N \times M$ . Imprima a matriz que foi fornecida.

```
matriz = []  
N = input("Número de linhas: ")  
M = input("Número de colunas: ")
```

```
#inicializando matriz – Prog. 1 – slide 13  
linha = [0]*M  
for i in range(N):  
    matriz = matriz+[linha]
```

```
#Preenchendo matriz  
for i in range(N):  
    for j in range (M):  
        mensagem="Entre com o elemento da linha "+ str(i+1)+" e coluna "+str(j+1)+" : "  
        matriz[i][j]= input(mensagem)
```

```
#Imprimindo matriz  
for i in range(N):  
    for j in range (M):  
        print matriz[i][j],  
    print
```

**ATENÇÃO: TESTE ESTE PROGRAMA**

# Listas - Matrizes

**Exercício:** Faça um programa que permita ao usuário construir uma matriz de inteiros  $N \times M$ . Imprima a matriz que foi fornecida.

```
matriz = []
N = input("Número de linhas: ")
M = input("Número de colunas: ")

#inicializando matriz – Prog. 2 – slide 13
for i in range(N):
    matriz.append([0]*M)

#Preenchendo matriz
for i in range(N):
    for j in range (M):
        mensagem="Entre com o elemento da linha "+ str(i+1)+" e coluna "+str(j+1)+" : "
        matriz[i][j]= input(mensagem)

#Imprimindo matriz
for i in range(N):
    for j in range (M):
        print matriz[i][j],
    print
```

**ATENÇÃO: TESTE ESTE PROGRAMA**

# Listas - Matrizes

## Exercício:

Escreva um programa para multiplicar os elementos da diagonal principal de uma dada matriz por um valor  $k$ .

Seu programa deve começar lendo o tamanho da matriz e em seguida ler os valores da matriz. Depois, leia o valor de  $k$ .

Imprima a matriz resultante.

# Listas - Matrizes

## Exercício:

Escreva um programa para multiplicar os elementos da diagonal principal de uma dada matriz por um valor k.

Seu programa deve começar lendo o tamanho da matriz e em seguida ler os valores da matriz. Depois, leia o valor de k.

Imprima a matriz resultante.

```
M1 = []
M1linha = input("Número de linhas e colunas da matriz : ")
#inicializando matriz
...
#Preenchendo matriz
...
# Entre com o valor de k
k = input("Valor para multiplicar a diagonal principal : ")

#Multiplicando diagonal por k
for i in range(M1linha):
    M1[i][i]=M1[i][i]*k

#Imprimindo matriz
for i in range(M1linha):
    for j in range (M1col):
        print M1[i][j],
    print
```

# Listas - Matrizes

## Exercício:

Escreva um programa para computar a linha de maior soma de uma matriz.

Seu programa deve começar lendo o tamanho da matriz e em seguida ler os valores da matriz.

Após determinar a linha de maior soma, esta linha deve ser impressa assim como sua soma.

# Listas - Matrizes

## Exercício:

Escreva um programa para computar a linha de maior soma de uma matriz.

Seu programa deve começar lendo o tamanho da matriz e em seguida ler os valores da matriz.

Após determinar a linha de maior soma, esta linha deve ser impressa assim como sua soma.

```
M1 = []
M1linha = input("Número de linhas da matriz : ")
M1col = input("Número de colunas da matriz : ")
#inicializando matriz
...
#Preenchendo matriz
...
#Calculando soma das linhas
maiorsoma=0
maiorlinha=1
for j in range(M1col):
    maiorsoma=maiorsoma+M1[0][j]
```

```
for i in range(1,M1linha):
    soma = 0
    for j in range(M1col):
        soma=soma+M1[i][j]
    if soma > maiorsoma:
        maiorsoma=soma
        maiorlinha=i+1

#Imprimindo matriz
for i in range(M1linha):
    for j in range (M1col):
        print M1[i][j],

print "linha de maior soma é %d com valor %d
      e elementos " % (maiorlinha,maiorsoma),
print M1[maiorlinha-1]
```

# Listas - Matrizes

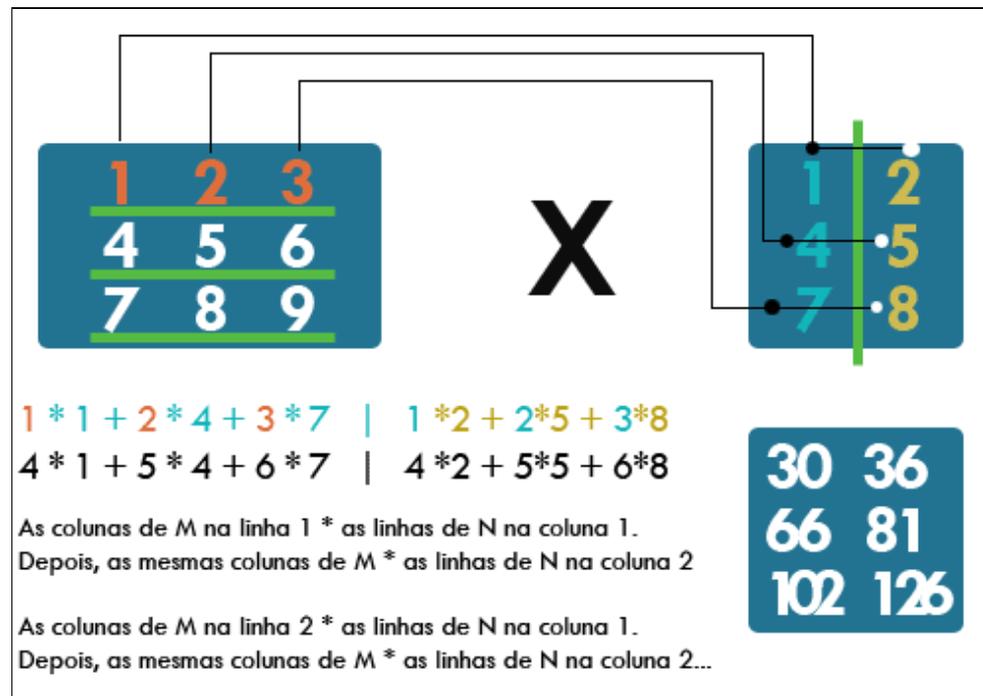
## Exercício:

Escreva um programa para computar o produto de duas matrizes m1 e m2.

Seu programa deve começar lendo os tamanhos das matrizes e verificando se são de tamanhos compatíveis para multiplicação.

Em seguida faça a leitura preenchendo os valores das matrizes m1 e m2.

Finalmente faça o cálculo matemático, imprimindo a matriz final no formato tradicional.



# **Aula 8 – Estrutura de Dados continuação**