



# Matrizes



# Matrizes

---

- ▶ Variável composta **multidimensional**

- ▶ É equivalente a um vetor, contudo permite a utilização de diversas dimensões acessadas via diferentes índices
- ▶ Pode ser pensada como um vetor onde cada célula é outro vetor, recursivamente
- ▶ Em diversas situações matrizes são necessárias para correlacionar informações



# Exemplo motivacional

---

- ▶ Assumindo que **um aluno é avaliado com cinco notas**, seria necessário um vetor de cinco posições para guardar as notas de um aluno...

	0	1	2	3	4
notas	10.0	7.0	9.0	5.5	6.0



# Exemplo motivacional

- ▶ Contudo, assumindo que **uma turma tem três alunos**, seria necessária uma matriz bidimensional para guardar as notas de todos os alunos de uma turma...

		notas				
		0	1	2	3	4
alunos	0	5.0	4.5	7.0	5.2	6.1
	1	2.1	6.5	8.0	7.0	6.7
	2	8.6	7.0	9.1	8.7	9.3

```
turma = [[5.0, 4.5, 7.0, 5.2, 6.1], [2.1, 6.5, 8.0, 7.0, 6.7],  
         [8.6, 7.0, 9.1, 8.7, 9.3]]
```

# Exemplo motivacional

- ▶ Contudo, assumindo que **uma turma tem três alunos**, seria necessária uma matriz bidimensional para guardar as notas de todos os alunos de uma turma...

		notas				
		0	1	2	3	4
alunos	0	5.0	4.5	7.0	5.2	6.1
	1	2.1	6.5	8.0	7.0	6.7
	2	8.6	7.0	9.1	8.7	9.3

```
turma= [[5.0, 4.5, 7.0, 5.2, 6.1], [2.1, 6.5, 8.0, 7.0, 6.7],  
        [8.6, 7.0, 9.1, 8.7, 9.3]]
```

# Exemplo motivacional

- ▶ Na verdade, na memória seria algo assim...

turma	0	0	5.0
		1	4.5
		2	7.0
		3	5.2
		4	6.1
	...	0	
		1	7.0
		2	9.1
		3	8.7
		4	9.3

# Acesso aos valores: [linha][coluna]

---

## ► Segunda nota do primeiro aluno

```
>>> turma[0][1]
```

```
4.5
```

## ► Quinta nota do terceiro aluno

```
>>> turma[2][4]
```

```
9.3
```

		notas				
		0	1	2	3	4
alunos	0	5.0	4.5	7.0	5.2	6.1
	1	2.1	6.5	8.0	7.0	6.7
	2	8.6	7.0	9.1	8.7	9.3

# Calcular a média da turma

---

```
turma = [[5.0, 4.5, 7.0, 5.2, 6.1],  
[2.1, 6.5, 8.0, 7.0, 6.7], [8.6, 7.0,  
9.1, 8.7, 9.3]]  
#calcula a média  
media = 0  
#for para percorrer as linhas  
for i in range(3):  
    #for para percorrer as colunas  
    for j in range(5):  
        media = media + turma[i][j]  
media = media / 15  
print(media)
```

---





# Preencher a matriz por leitura

---

```
turma = []
for i in range(3):
    # cria linha vazia
    linha = []
    for j in range(5):
        #vai adicionando as notas na linha
        linha.append(int(input('Digite a nota
[' + str(i) + ', ' + str(j) + ']:')))
    #adiciona a linha na matriz turma
    turma.append(linha)
```



# Exemplo

---

- ▶ Programa que cria uma matriz **n x m** preenchida com zeros

```
n = int(input('Digite a dimensão n da matriz: '))
m = int(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    linha = []
    for j in range(m):
        linha.append(0)
    matriz.append(linha)
print(matriz)
```



# Simplificando o exemplo

---

- ▶ Programa que cria uma matriz **n x m** preenchida com zeros

```
n = int(input('Digite a dimensão n da matriz: '))
m = int(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    matriz.append([0]*m)
print(matriz)
```



# Imprimir em forma de matriz

---

- ▶ Programa que cria uma matriz **n x m** preenchida com zeros e a **imprime no formato de matriz**

```
n = int(input('Digite a dimensão n da matriz: '))
m = int(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    matriz.append([0]*m)
#imprimir em formato de matriz
for i in range(n):
    print(matriz[i])
```



# Imprimir em forma de matriz

---

- ▶ Programa que cria uma matriz **n x m** preenchida com zeros e a **imprime no formato de matriz**

```
n = int(input('Digite a dimensão n da matriz: '))
m = int(input('Digite a dimensão m da matriz: '))
matriz = []
for i in range(n):
    matriz.append([0]*m)
#imprimir em formato de matriz
for i in range(n):
    print(matriz[i])
```

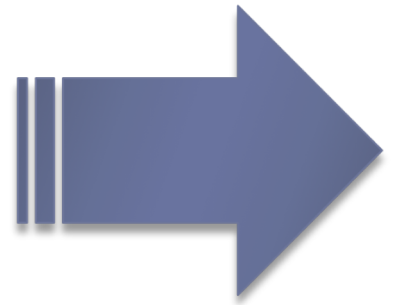
Resultado para matriz 2x3:

```
[0, 0, 0]
[0, 0, 0]
```

# Exemplo Contar Pares

---

- ▶ Programa que lê uma matriz 3x3 digitada pelo usuário e conta quantos números pares existem na matriz, imprimindo na tela o resultado e a matriz.



# Exemplo Contar Pares

---

```
matriz = []
for i in range(3):
    linha = []
    for j in range(3):
        linha.append(int(input('Digite o valor de ['
+ str(i) + ', ' + str(j) + ']:')))
    matriz.append(linha)
#contar pares
pares = 0
for i in range(3):
    for j in range(3):
        if matriz[i][j] % 2 == 0:
            pares = pares + 1
#imprimir em formato de matriz
for i in range(3):
    print(matriz[i])
#imprimir qtde de números pares
print('A matriz contém', pares, 'números pares')
```



# Relembrando: for iterando sobre valores

---

- ▶ Um comando for também pode iterar sobre valores de uma lista

```
lista = [1, 2, 4, 5, 7, 8, 9]
for i in lista:
    print(i)
```





# Variação do Exemplo Contar Pares

---

```
matriz = []
for i in range(3):
    linha = []
    for j in range(3):
        linha.append(int(input('Digite o valor de [' +
str(i) + ', ' + str(j) + ']:')))
    matriz.append(linha)
#contar pares
pares = 0
for linha in matriz:
    for valor in linha:
        if valor % 2 == 0:
            pares = pares + 1
#imprimir em formato de matriz
for i in range(3):
    print(matriz[i])
#imprimir qtde de números pares
print('A matriz contém', pares, 'números pares')
```

# Dimensões da matriz

---

- ▶ É possível usar `len()` para saber a dimensão de uma matriz
- ▶ Assumindo que todas as linhas possuem o mesmo número de elementos, pode-se fazer:
  - ▶ Número de linhas: `len(matriz)`
  - ▶ Número de colunas: `len(matriz[0])`

```
matriz = []
parar = False
while not(parar):
    linha = [0] * 10
    matriz.append(linha)
    x = input("Deseja parar? (S/N) ")
    if x == "S":
        parar = True
print("A matriz possui %d linhas" % len(matriz))
print("A matriz possui %d colunas" % len(matriz[0]))
```

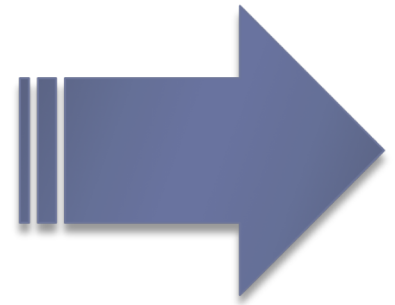


# Python permite misturar tipos em uma matriz

---

- ▶ Exemplo: programa que armazena os nomes e idades de 10 pessoas em uma matriz, e imprime o nome da pessoa mais nova

Ana	10
Lucas	15
Bia	13
Larissa	24
Leo	21
Bruno	32
Cássio	4
Jonas	8
Lauro	23
Mateus	18



# Encontra a pessoa mais nova

---

```
m = []
#preenche a matriz
for i in range(10):
    linha = []
    linha.append(input('Digite o nome da pessoa ' +
str(i) + ':'))
    linha.append(int(input('Digite a idade de ' +
linha[0] + ':')))
    m.append(linha)
#procura a pessoa mais nova
menor = m[0][1]
pos = 0
for i in range(10):
    if m[i][1] < menor:
        menor = m[i][1]
        pos = i
#imprime a matriz
for i in range(10):
    print(m[i])
print('A pessoa mais nova é', m[pos][0])
```

Ana	10
Lucas	15
Bia	13
Larissa	24
Leo	21
Bruno	32
Cássio	4
Jonas	8
Lauro	23
Mateus	18



# Matrizes

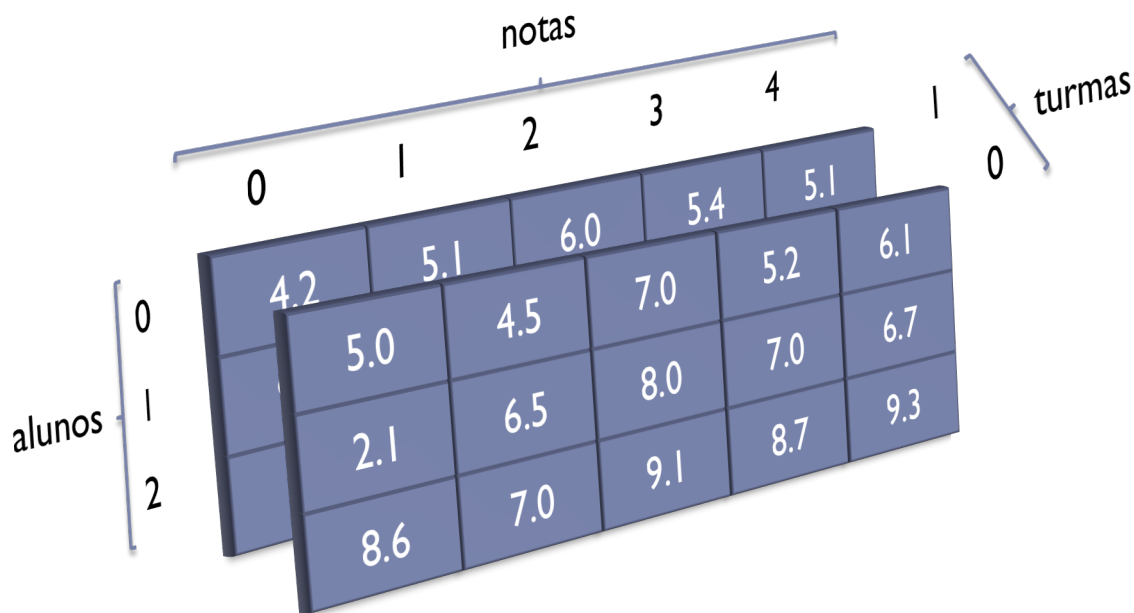
---

- ▶ Uma matriz pode ter um número qualquer de dimensões!  
Basta usar um índice para cada dimensão.



# Exemplo motivacional

- ▶ Ainda, assumindo que **um curso tem duas turmas**, seria necessária uma matriz tridimensional para guardar as notas de todos os alunos de todas as turmas do curso.



# Atribuição

```
>>> m = [[5.0, 4.5, 7.0, 5.2,  
5.1], [2.1, 6.5, 8.0, 7.0, 6.7], [8.6, 7.0, 9.1, 8.7, 9.3]], [[  
4.2, 5.1, 6.0, 5.4, 5.1], [9.0, 8.0, 7.5, 8.1, 8.8], [2.3, 4.4,  
6.7, 6.6, 7.0]]]
```

**Turma**

		notas				
		0	1	2	3	4
alunos	0	4.2	5.1	6.0	5.4	5.1
	1	5.0	4.5	7.0	5.2	6.1
	2	2.1	6.5	8.0	7.0	6.7
		turmas				
		0				
		0	8.6	7.0	9.1	8.7
		1	9.0	8.0	7.5	8.1
		2	2.3	4.4	6.7	6.6

# Atribuição

```
>>> m = [[[5.0, 4.5, 7.0, 5.2,  
5.1], [2.1, 6.5, 8.0, 7.0, 6.7], [8.6, 7.0, 9.1, 8.7, 9.3]], [[  
4.2, 5.1, 6.0, 5.4, 5.1], [9.0, 8.0, 7.5, 8.1, 8.8], [2.3, 4.4,  
6.7, 6.6, 7.0]]]
```

**Aluno**

		notas						turmas
		0	1	2	3	4		
alunos	0	4.2	5.1	6.0	5.4	5.1		0
	1	5.0	4.5	7.0	5.2	6.1		
	2	2.1	6.5	8.0	7.0	6.7		
		8.6	7.0	9.1	8.7	9.3		



# Acesso a elemento

```
>>> m = [[[5.0, 4.5, 7.0, 5.2,  
5.1], [2.1, 6.5, 8.0, 7.0, 6.7], [8.6, 7.0, 9.1, 8.7, 9.3]], [[  
4.2, 5.1, 6.0, 5.4, 5.1], [9.0, 8.0, 7.5, 8.1, 8.8], [2.3, 4.4,  
6.7, 6.6, 7.0]]]
```

```
>>> print(m[0][1][0])
```

**2.1**

**Turma**  
**Aluno**  
**Nota**

		notas					
		0	1	2	3	4	
alunos	0	4.2	5.1	6.0	5.4	5.1	turmas 0
	1	5.0	4.5	7.0	5.2	6.1	
	2	2.1	6.5	8.0	7.0	6.7	
		0	1	2	3	4	
		8.6	7.0	9.1	8.7	9.3	

# Exercícios

---

1. Faça um programa que leia uma matriz  $3 \times 3$  de inteiros e multiplique os elementos da diagonal principal da matriz por um número  $k$ . Imprima a matriz na tela antes e depois da multiplicação.
2. Faça um programa que leia duas matrizes  $A$  e  $B$   $2 \times 2$  de inteiros e imprima a matriz  $C$  que é a soma das matrizes  $A$  e  $B$ .
3. Faça um programa que leia as dimensões de duas matrizes  $A$  e  $B$ , e depois leia as duas matrizes (os elementos devem ser inteiros). Se as matrizes forem de tamanhos compatíveis para multiplicação, multiplique as matrizes. Imprima as matrizes  $A$ ,  $B$  e a matriz resultante da multiplicação.



# Exercícios

---

4. Faça um programa que leia uma matriz 3x3 de inteiros e retorne a linha de maior soma. Imprima na tela a matriz, a linha de maior soma e a soma.
5. Faça um programa que leia a ordem de uma matriz quadrada A (até 100), posteriormente leia seus valores e escreva sua transposta AT, onde  $AT[i][j] = A[j][i]$
6. Uma pista de Kart permite 10 voltas para cada um de 6 corredores. Faça um programa que leia os nomes e os tempos (em segundos) de cada volta de cada corredor e guarde as informações em uma matriz. Ao final, o programa deve informar:
  - a. De quem foi a melhor volta da prova, e em que volta
  - b. Classificação final em ordem (1º. o campeão)
  - c. Qual foi a volta com a média mais rápida



# Exercícios

---

7. Faça um programa que leia uma matriz  $6 \times 3$  com números reais, calcule e mostre: (a) o maior elemento da matriz e sua respectiva posição (linha e coluna); (b) o menor elemento da matriz e sua respectiva posição.
8. Faça um programa que leia duas matrizes A e B de números inteiros e verifique se ambas são inversas (ou seja, se a multiplicação de A por B é a matriz identidade).
9. Faça um programa que leia uma matriz  $3 \times 3$  que representa um tabuleiro de jogo da velha e indique qual posição deveria ser jogada para ganhar o jogo (se possível) ou ao menos evitar uma derrota.



# Exercícios

---

10. Faça um programa que lê duas notas para cada aluno de duas turmas. Cada turma tem 3 alunos. Armazene os dados em uma matriz  $M$ . Cada aluno deve ter três notas (as duas digitadas e a média dessas duas). Calcule a média de cada turma e armazene em um vetor  $TURMA$ . Informe qual turma tem maior média, e quais alunos tiveram média maior que a média de sua turma.



# Referências

---

- ▶ Slides baseados no curso de Leonardo Murta e Aline Paes