

#### Universidade Federal de Ouro Preto - UFOP Departamento de Computação - DECOM Programação de Computadores I – BCC701



# Aula Teórica 09

**Laços Aninhados** 



# Universidade Federal de Ouro Preto - UFOP Departamento de Computação - DECOM Programação de Computadores I – BCC701



- P dir(\_\_builtins\_\_) Mostra as funções disponíveis no Python. Podem ser usadas diretamente no programa. Ex.: abs(), input(), print(), int(), len(), pow, str()...
- dir(str): mostra os métodos contidos no str: str.upper(), str.lower(), ...
- help(str.upper): mostra como usar e o que produz o
  método

As funções padrão do Python são diferentes de métodos contidos em determinados pacotes como o math.sqrt()do math

- dir(math) exige que o pacote math seja incluído: import
  math e mostra os métodos disponíveis neste pacote.
- help(math.ceil)



- Em muitos casos temos a necessidade de usar um laço dentro do corpo de um outro laço.
- Situações como esta aparecem sempre que estamos trabalhando com uma tabela, matrizes entre outras.
- Para percorrer uma tabela temos que percorrer as linhas e para cada linha, percorrer as suas colunas.

	↓ j = 1	↓ j = 2	↓ j = 3	↓ j = 4
→ i = 1	a11	a12	a13	a14
1-1	a21	a22	a23	a24
	a31	a32	a33	a34



- Em muitos casos temos a necessidade de usar um laço dentro do corpo de um outro laço.
- Situações como esta aparecem sempre que estamos trabalhando com uma tabela, matrizes entre outras.
- Para percorrer uma tabela temos que percorrer as linhas e para cada linha, percorrer as suas colunas.

	↓ j = 1	↓ j = 2	↓ j = 3	↓ j = 4
	a11	a12	a13	a14
<b>→</b> = 2	a21	a22	a23	a24
	a31	a32	a33	a34



- Em muitos casos temos a necessidade de usar um laço dentro do corpo de um outro laço.
- Situações como esta aparecem sempre que estamos trabalhando com uma tabela, matrizes entre outras.
- Para percorrer uma tabela temos que percorrer as linhas e para cada linha, percorrer as suas colunas.

<b>↓</b> j = 1	↓ j = 2	↓ j = 3	↓ j = 4
a11	a12	a13	a14
a21	a22	a23	a24
a31	a <b>32</b>	a <b>33</b>	a <b>34</b>

→ i = 3

Observe que para cada posição/elemento, temos 2 índices:

- linha i
- coluna j



a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34

```
for lin in range(1, 4):
    print(f"linha {lin}")
```

### Saída:

linha 1

linha 2

linha 3



a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34

```
for lin in range (1, 4):
    print(f"linha{lin}:")
    for col in range(1, 5):
        print("coluna {col}")
    print(f" :: fim da linha {lin}")
```

```
Saída:
```

linha 1 : col. 1 col. 2 col. 3 col. 4 :: fim da linha 1 linha 2 : col. 1 col. 2 col. 3 col. 4 :: fim da linha 2 linha 3 : col. 1 col. 2 col. 3 col. 4 :: fim da linha 3



a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





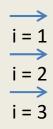
a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34



a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





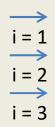
a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34





a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34







a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34

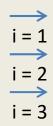




a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34







a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34







a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34



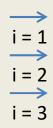




a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34







a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34



a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34

```
for col in range(1, 5):
    print(f"coluna {col}")
```

### Saída:

coluna 1

coluna 2

coluna 3

coluna 4



a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34

```
for col in range(1, 5):
    print(f"coluna %g : {col}")
    for lin in range(1, 4):
        print(f"linha {lin}")
    print(f" :: fim da coluna {col}")
```

#### Saída:

coluna 1 : linha 1 linha 2 linha 3 :: fim da coluna 1 coluna 2 : linha 1 linha 2 linha 3 :: fim da coluna 2 coluna 3 : linha 1 linha 2 linha 3 :: fim da coluna 3 coluna 4 : linha 1 linha 2 linha 3 :: fim da coluna 4



Um casal que tem 3 filhos quer contabilizar os seus gastos com os filhos ao longo dos 4 trimestres do ano. Cada filho tem um gasto diferente, então foi feita a tabela abaixo para ajudar o casal neste controle.

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00

Vamos fazer um programa simplesmente para ler/entrar com os dados da tabela, usando laços for.

Esta leitura pode ser feita de duas maneiras diferentes, por linha ou por coluna.

Inicialmente vamos fazer a leitura percorrendo a Tabela por linha.



```
print('Digite o gasto por filho para cada trimestre')
```

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
```

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



```
print('Digite o gasto por filho para cada trimestre')
```

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    # percorre os trimestres
    for tri in range(1, 5):
        gasto = float(input(f'\tGasto no trimestre ({tri}): '))
```

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por linha

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    # percorre os trimestres
    for tri in range(1, 5):
        gasto = float(input(f'\tGasto no trimestre ({tri}): '))
```

$$fil = 1, tri = 1, 2, 3, 4$$

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por linha

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    # percorre os trimestres

for tri in range(1, 5):
    gasto = float(input(f'\tGasto no trimestre ({tri}): '))
```

$$fil = 2, tri = 1, 2, 3, 4$$

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por linha

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    # percorre os trimestres
    for tri in range(1, 5):
        gasto = float(input(f'\tGasto no trimestre ({tri}): '))
```

$$fil = 3$$
,  $tri = 1$ , 2, 3, 4

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



Um casal tem 3 filhos e quer contabilizar os seus gastos com o filhos ao longo dos 4 trimestres do ano. Cada filho tem um gasto diferente, então foi feita a tabela abaixo para ajudar o casal neste controle.

Vamos fazer um programa simplesmente para <u>ler/entrar com os dados da tabela</u>, usando laços for.

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00

Agora vamos fazer a leitura percorrendo a Tabela por coluna



#### Percorrendo a tabela por coluna

```
# percorre cada trimestre
for tri in range(1, 5):
    print(f' === Trimestre ({tri}) === ') # informa o trimestre
```

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por coluna

```
# percorre cada trimestre
for tri in range(1, 5):
    print(f' === Trimestre ({tri}) === ') # informa o trimestre
    # percorre os filhos
    for fil in range(1, 4):
        gasto = float(input(f'\tGasto do filho ({fil}): '))
```

$$tri = 1, fil = 1, 2, 3$$

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por coluna

```
# percorre cada trimestre
for tri in range(1, 5):
    print(f' === Trimestre ({tri}) === ') # informa o trimestre
    # percorre os filhos
    for fil in range(1, 4):
        gasto = float(input(f'\tGasto do filho ({fil}): '))
```

$$tri = 2, fil = 1, 2, 3$$

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por coluna

```
# percorre cada trimestre
for tri in range(1, 5):
    print(f' === Trimestre ({tri}) === ') # informa o trimestre
    # percorre os filhos
    for fil in range(1, 4):
        gasto = float(input(f'\tGasto do filho ({fil}): '))
```

$$tri = 3, fil = 1, 2, 3$$

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



#### Percorrendo a tabela por coluna

print('Digite o gasto por filho para cada trimestre')

```
# percorre cada trimestre
for tri in range(1, 5):
    print(f' === Trimestre ({tri}) === ') # informa o trimestre
    # percorre os filhos
    for fil in range(1, 4):
        gasto = float(input(f'\tGasto do filho ({fil}): '))
```

$$tri = 4$$
,  $fil = 1$ , 2, 3

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4	
Filho 1	1.250,00	2.000	1.200	1.850,00	
Filho 2	1.000,00	1.200	850,00	1.500,00	
Filho 3	850,00	1.000	750,00	1.250,00	



Quando temos um laço dentro de outro chamamos de laços aninhados: Suponha n = 2 e m = 3

i	j	(i, j)
1	1	(1,1)
1	2	(1,2)
1	3	(1,3)
2	1	(2,1)
2	2	(2,2)
2	3	(2,3)

i j	1	2	3
1	(1,1)	(1,2)	(1,3)
2	(2,1)	(2,2)	(2,3)



Quando temos um laço dentro de outro temos laços aninhados:

```
n = int(input("..."))
m = int(input("..."))
```

- A execução começa no laço externo (azul), fixando o valor de i;
- Quando chegamos ao laço interno (vermelho), suas m interações são realizadas (j varia entre 1 e m) com o i fixo;
- Ao sair do laço mais interno, incrementa-se o contador do laço externo, i.
- 2. Se ocorrer a repetição do bloco do laço externo, o laço interno será executado novamente.



Um casal tem 3 filhos e quer contabilizar os seus gastos com o filhos ao longo dos 4 trimestres do ano. Cada filho tem um gasto diferente, então foi feita a tabela abaixo para ajudar o casal neste controle.

Vamos fazer um programa para ler/entrar com os dados da tabela, <u>somar e</u> <u>informar o gasto com cada filho</u> no ano todo.

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00

Para somar os valores de cada filho (linha), temos que percorrer a tabela por linhas!

#### Percorrendo a tabela por linha



print('Digite o gasto por filho para cada trimestre')

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    # percorre os trimestres

for tri in range(1, 5):
    gasto = float(input(f'\tGasto no trimestre ({tri}): '))
```

Temos que percorrer a tabela por linhas e somar os elementos da linha 1, depois, somar os elementos da linha 2 e finalmente somar os elementos da linha 3.

Assim, temos 3 somas independentes. Precisamos de uma variável acumuladora e zerá-la antes de começar a soma de cada filho

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4	
Filho 1	1.250,00	2.000	1.200	1.850,00	
Filho 2	1.000,00	1.200	850,00	1.500,00	
Filho 3	850,00	1.000	750,00	1.250,00	



# Temos 3 somas independentes, uma para cada filho. A variável acumuladora deve ser zerada antes de começar a soma de cada filho

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00

#### Onde zerar a variável acumuladora soma?



#### Percorrendo a tabela por linha

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4	
Filho 1	1.250,00	2.000	1.200	1.850,00	
Filho 2	1.000,00	1.200	850,00	1.500,00	
Filho 3	850,00	1.000	750,00	1.250,00	



# Temos 3 somas independentes, uma para cada filho. A variável acumuladora deve ser zerada antes de começar a soma de cada filho

#### Percorrendo a tabela por linha

```
print('Digite o gasto por filho para cada trimestre')

Soma = 0 aqui?  #estamos zerando uma única vez => soma geral

# percorre cada filho

for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho

    Soma = 0 aqui? #local correto, zerando antes de iniciar cada linha

# percorre os trimestres

for tri in range(1, 5):
    Soma = 0 ou aqui?  #zerando a cada leitura => não soma

    gasto = float(input(f'\tGasto no trimestre ({tri}): '))
    soma = soma + gasto
```

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4	
Filho 1	1.250,00	2.000	1.200	1.850,00	
Filho 2	1.000,00	1.200	850,00	1.500,00	
Filho 3	850,00	1.000	750,00	1.250,00	



# Temos 3 somas independentes, uma para cada filho. A variável acumuladora deve ser zerada antes de começar a soma de cada filho

#### Percorrendo a tabela por linha

print("Fim do programa")

```
print('Digite o gasto por filho para cada trimestre')
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    soma = 0
# percorre os trimestres

for tri in range(1, 5):
    gasto = float(input(f'\tGasto no trimestre ({tri}): '))
    soma = soma + gasto
# impressão do gasto por filho fora do laço dos trimestres
print(f"Gasto com o filho {fil}: R${soma:8.2f}")
```

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4	
Filho 1	1.250,00	2.000	1.200	1.850,00	
Filho 2	1.000,00	1.200	850,00	1.500,00	
Filho 3	850,00	1.000	750,00	1.250,00	

#### Percorrendo a tabela por linha

print("Fim do programa")



```
print('Digite o gasto por filho para cada trimestre')
soma_geral = 0
```

```
# percorre cada filho
for fil in range(1, 4):
    print(f' === Filho ({fil}) === ') # informa o filho
    soma = 0

# percorre os trimestres
for tri in range(1, 5):
    gasto = float(input(f'\tGasto no trimestre ({tri}): '))
    soma = soma + gasto
    print(f"Gasto com o filho {fil}: R${soma:8.2f}")
    soma_geral += soma

print(f"Gasto total com os 3 filhos: R${soma geral:8.2f}")
```

Uma outra possibilidade seria verificar os gastos por trimestre. Fazer o mesmo, mas agora por coluna!

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.200	850,00	1.500,00
Filho 3	850,00	1.000	750,00	1.250,00



Um casal tem 3 filhos e quer contabilizar os seus gastos com o filhos ao longo dos 4 trimestres do ano. Cada filho tem um gasto diferente, então foi feita a tabela abaixo para ajudar o casal neste controle.

Vamos fazer um programa simplesmente para <u>ler/entrar com os dados da</u> <u>tabela</u>, usando laços for.

	Trimestre 1	Trimestre 2	Trimestre 3	Trimestre 4
Filho 1	1.250,00	2.000	1.200	1.850,00
Filho 2	1.000,00	1.500	850,00	1.200,00
Filho 3	1.300,00	1.000	750,00	850,00

Nesta situação podemos implementar solução para as seguintes questões:

- 1. qual é o filho com o maior gasto anual?
- 2. qual é o trimestre com maior gasto com os 3 filhos?
- 3. qual é o filho com o maior gasto em cada trimestre?
- 4. qual é o trimestre em que cada filho gastou mais?

## Exercício – Tabuada de Multiplicação



# Faça um programa que imprima a tabela da tabuada de multiplicação:

	1	2	3	4	5	6	7	8	9	10
1	1	2	3	4	5	6	7	8	9	10
2	2	4	6	8	10	12	14	16	18	20
3	3	6	9	12	15	18	21	24	27	30
4	4	8	12	16	20	24	28	32	36	40
5	5	10	15	20	25	30	35	40	45	50
6	6	12	18	24	30	36	42	48	54	60
7	7	14	21	28	35	42	49	56	63	70
8	8	16	24	32	40	48	56	64	72	80
9	9	18	27	36	45	54	63	72	81	90
10	10	20	30	40	50	60	70	80	90	100

#### Exercício – Tabuada de Multiplicação



```
print("Programa da Tabuada")
for lin in range(1, 11):
    print(f"linha {lin}: ", end = "")
    for col in range(1, 11):
        print(f" {lin*col} ", end = "")
    print(f"")
print("Fim da Tabuada")
```

Acrescentar formatação

#### Exercício – Tabuada de Multiplicação





# **Exercícios**

- Faça um programa para calcular e imprimir a média de cada aluno de uma série de 10 exercícios feitos em uma disciplina que conta com 35 alunos matriculados.
- Agora acrescente uma maneira de calcular a média das médias da turma toda para os exercícios
- Obs. Resolução a seguir



```
Soma_geral = 0
for alu in range(1, 36):
       soma = 0
      print(f"aluno {alu}")
       for ex in range(1, 11):
             nt = float(input("exercício {ex}: "))
             soma = soma + nt
      md = soma/10
      print(f"média : {md:5.2f}")
       soma geral = soma geral + md
print(f"Média da turma: {soma_geral/35:5.2f}")
```