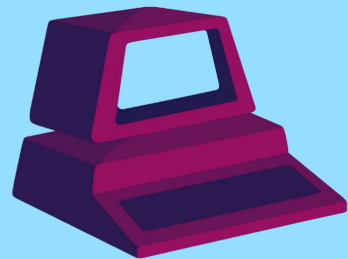


Aula 4 - Turma Avançado

- Estruturas de Repetição

PET



COMPUTAÇÃO

2026

1.

Entendendo as Estruturas

Loops

- Estruturas de repetição, que chamamos de laços ou loops, permitem a execução um ou mais comandos quantas vezes desejarmos.
- Isso é muito útil para percorrer uma determinada quantidade de dados, evitando repetição de código.
- Geralmente, temos dois tipos de loops: **while** e **for**.

Loops

- Exemplo de código: soma dos números de 1 a 5

Sem loop:

```
soma = 0
i = 1

soma = soma + i
i = i + 1
soma = soma + i
i = i + 1
soma = soma + i
i = i + 1
soma = soma + i
i = i + 1
soma = soma + i

print(soma)
```

Com loop:

```
soma = 0
i = 1
while (i <= 5):
    soma = soma + i
    i = i + 1
print(soma)
```

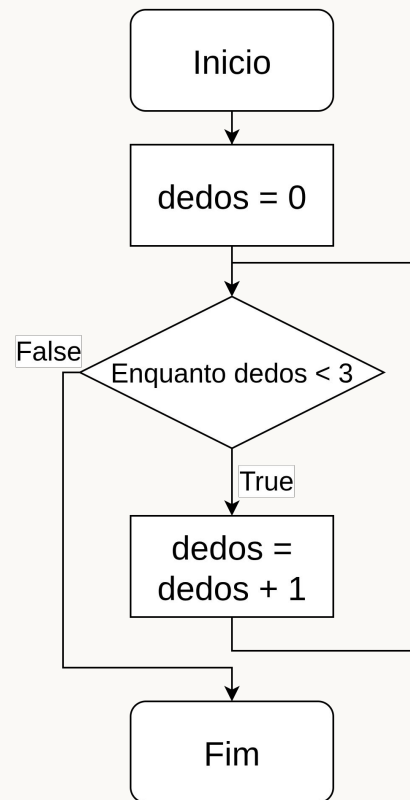
Loops - While

“While”, em inglês, significa “Enquanto”.

Usamos essa estrutura para repetir uma ação **enquanto** uma certa condição for verdadeira.

Ex: Contar até três:

- `dedos = 0`
- `enquanto dedos < 3, dedos = dedos + 1`



Lembrando - Operadores de comparação

- == — igual
- != — diferente
- > — maior que
- < — menor que
- >= — maior ou igual que
- <= — menor ou igual que

O resultado dessas operações será **verdadeiro** ou **falso**!



```
1 comparacao = a == b
2 comparacao = a != b
3 comparacao = a > b
4 comparacao = a < b
5 comparacao = a >= b
6 comparacao = a <= b
```

Loops - While

Exemplo de While em Python:



```
1  dedos = 0
2  while dedos < 3:
3      dedos = dedos + 1
```

OBS: A variável “dedos” é chamada **variável de controle**, através dela que o while decide se vai iterar (continuar) o loop ou não.

Loops - For

For, em inglês, significa “Para”. Ou seja, o comando será executado **para cada valor em um determinado intervalo**.

Exemplo: soma dos números de 1 até 5

Com while:

```
soma = 0
i = 1
while (i <= 5):
    soma = soma + i
    i = i + 1
print(soma)
```

Com for:

```
soma = 0
for i in range(1, 6):
    soma = soma + i
print(soma)
```

- OBS: aqui é pra ser um intervalo de 1 até 6 mesmo (já falamos sobre)
- Range é uma função que significa intervalo em inglês.

Loops - For

- While: enquanto i for menor ou igual a 5, adicione i ao valor total da soma.
- For: para cada i entre 1 e 6, sem incluir o próprio 6, adicione i ao valor total da soma.

O range funciona dessa forma por conta das listas, que estudaremos nas próximas aulas.

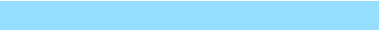
Com while:

```
soma = 0
i = 1
while (i <= 5):
    soma = soma + i
    i = i + 1
print(soma)
```

Com for:

```
soma = 0
for i in range(1, 6):
    soma = soma + i
print(soma)
```

Função range()



A função range() pode ser utilizada de 3 formas:

- range(fim): seleciona valores de 0 até fim
- range(início, fim): seleciona valores de início até fim
- range(início, fim, passo): seleciona valores de início até fim, somando de passo em passo. Ex: range(1, 10, 2) = 1, 3, 5, 7, 9.

Loops - For

A diferença entre o 'while' e o 'for' é esta:

- Para o while, dependemos de uma condição ser verdadeira ou falsa para a repetição parar ou continuar. Então, não sabemos exatamente quantas vezes um comando vai ser executado, pois isso só depende da condicional.
- Já no caso do for, sabemos exatamente quantas vezes o comando em repetição vai ser executado. O começo e o fim do loop já são pré-estabelecidos, conforme um intervalo.

Loops - For

Na prática, o que está por trás do 'for' é um 'while'.

- Toda repetição feita com um 'for' poderia ser feita usando 'while'.
- Mas nem toda repetição feita com um 'while' poderia ser feita usando 'for'.

Loops - For

Exemplo de For em Python:



```
1 nome = "roberto"
2
3 for i in range(0, 7):
4     print(nome[i])
```

Traduzindo:

- “Enquanto a variável *i* estiver no intervalo de 0 a 7”
- Imprima na tela o cada letra do nome

Indentação

- *Relembrando:* a indentação é o uso de espaços ou tabulações no início de uma linha para criar blocos de código. É fundamental para a legibilidade do código, agindo como a pontuação de um texto.



```
1  # Código com erro de indentação
2  idade = 18
3  if idade >= 18:
4  print("Você pode dirigir.")
5  print("Aproveite o passeio!")
```



```
1  # Código com indentação correta
2  idade = 18
3  if idade >= 18:
4      print("Você pode dirigir.")
5      print("Aproveite o passeio!")
```

Indentação

- A indentação define o bloco de comandos a serem executados repetidamente.

Indentação correta

```
soma = 0
i = 1
while (i <= 5):
    soma = soma + i
    i = i + 1
print(soma)
```

Indentação incorreta

```
soma = 0
i = 1
while (i <= 5):
soma = soma + i
i = i + 1
print(soma)
```

2.

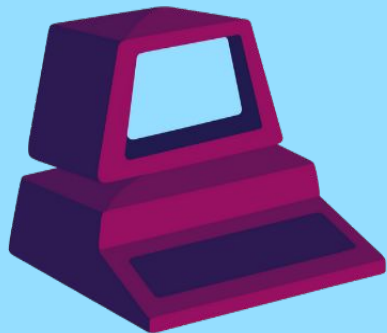
**Trabalhando com as
Estruturas de Repetição**

Exercícios

Usando **estruturas de repetição**

1. Imprima Números de 10 a 1 (nessa ordem).
2. Calcular a Soma dos Números de 1 a 100.
3. Imprima a tabuada do 3.

Obrigado!



PET
COMPUTAÇÃO

[pet.inf.ufpr.br](mailto:pet@inf.ufpr.br)
pet@inf.ufpr.br
[@petcompufpr](https://twitter.com/petcompufpr)