



## Pró-reitora de Pós-Graduação, Pesquisa e Inovação Especialização em Ciências de Dados e Analytics

Programação para Ciência de Dados  
Aula 2

Prof. Rodrigo Cesar Lira da Silva  
[rcls@ecomp.poli.br](mailto:rcls@ecomp.poli.br)



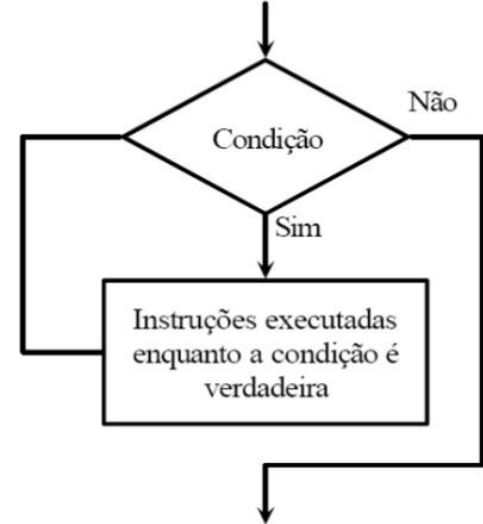
# Estruturas de Repetição

# Objetivos

- Apresentar o conceito de estruturas de Repetição em Python

# Estrutura de Repetição

- Repetições representam a base de vários programas.
- São utilizadas para executar a mesma parte de um programa várias vezes, normalmente dependendo de uma condição.



# Estrutura de Repetição

- Por exemplo, um programa que imprima de 1 a 3 na tela.

```
x = 1  
print (x)  
x += 1  
print (x)  
x += 1  
print (x)
```

- Imagine escrever de 1 a 100 na tela

# Estrutura de Repetição

- O exemplo da impressão dos números usando o while ficaria:



```
1 x = 1
2 while x <= 3:
3     print(x)
4     x = x + 1
```



```
1
2
3
```

# Estrutura de Repetição

- A estrutura while repete um bloco de código enquanto a condição for verdadeira.

**while** <condição>:  
*bloco*

- A condição é uma expressão lógica, e o bloco representa linhas dos programas a repetir

# Exercício

- ❏ Modifique a impressão de números apresentada anteriormente para imprimir de 1 até 100.

```
1 x = 1
2 while x <= 3:
3     print(x)
4     x = x + 1
```

---

- ❏ Modifique o programa anterior para imprimir de 50 até 100.



# Contadores

- ❑ O poder das estruturas de repetições é muito interessante, principalmente quando utilizamos condições com mais de uma variável.
- ❑ Normalmente utilizamos variáveis para controlar a quantidade de repetições


```
fim = int(input("Digite o último número: "))  
x = 1  
while x <= fim:  
    print (x)  
    x += 1
```

# Aninhamento de Estruturas

- É possível combinar diferentes estruturas

```
pontos = 0
questao = 1
while questao <= 3:
    resposta = input(f"Resposta da questão {questao}")
    if questao == 1 and resposta == "b":
        pontos = pontos + 1
    if questao == 2 and resposta == "a":
        pontos = pontos + 1
    if questao == 3 and resposta == "d":
        pontos = pontos + 1
    questao = questao + 1
print (f"O aluno fez {pontos} ponto(s)")
```

```
if questao == 3 and resposta == "d":  
    pontos = pontos + 1  
    questao = questao + 1  
print (f"Você fez {pontos} ponto(s)")
```



Operador	Exemplo	Equivalência
+=	x += 1	x = x + 1
-=	x -= 1	y = y - 1
*=	c *= 2	c = c * 2
/=	d /= 2	d = d / 2
**=	e **= 2	e = e ** 2
//=	f //= 4	f = f // 4

# Instrução Break/Continue

- Interrompendo uma instrução com break

```
soma = 0
while True:
    valor = int(input("Digite um número a somar ou 0 para sair:"))
    if valor == 0:
        break
    soma += valor
print (soma)
```

- E o **continue**?
  - Interrompe o fluxo da iteração atual e inicia a próxima iteração.

# Repetições Aninhadas

```
tabuada = 1
while tabuada <= 10:
    numero = 1
    while numero <= 10:
        print (f"{tabuada} x {numero} = {tabuada * numero}")
        numero += 1
    tabuada += 1
```