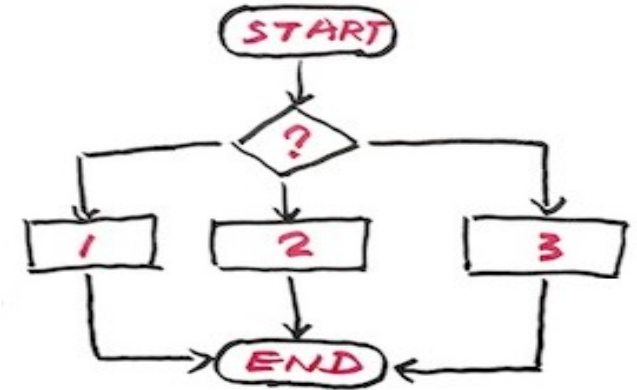


Introdução à Programação

Aula 08
Estrutura de repetição
While (Enquanto)



Prof. Dr. Ivan José dos Reis Filho

Aula passada

Estrutura de Repetição

- for

Aula de hoje

Objetivos

- Apresentar a estrutura de repetição **While (Enquanto)**

Enquanto

Repetição

Enquanto

Nesta estrutura, uma expressão lógica é avaliada. Se a mesma for verdadeira, um bloco de comandos será executada;

A execução da lista **se repetirá** até que a avaliação da condição resulte em **FALSO**;

Estrutura de repetição

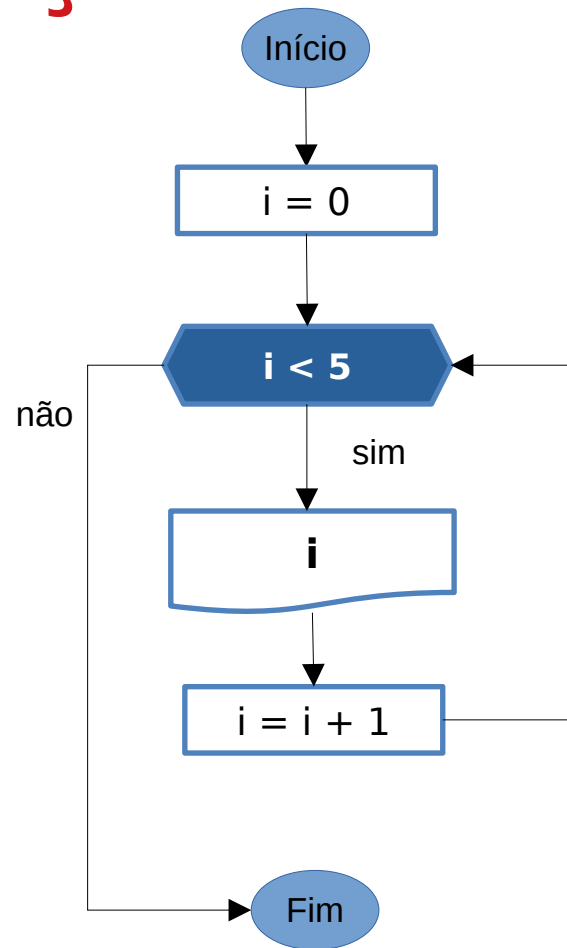
Tipos

For

- Define **explicitamente** a quantidade de vezes [ciclos] que será executado o bloco de instruções;

While

- Repete um bloco de instrução **enquanto** a condição definida no seu cabeçalho for verdadeira.



Estrutura de repetição

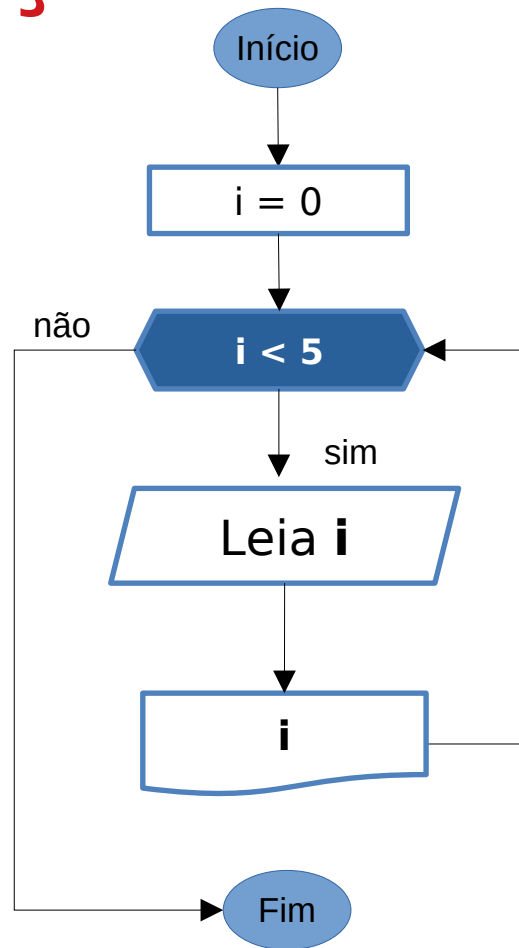
Tipos

For

- Define **explicitamente** a quantidade de vezes [ciclos] que será executado o bloco de instruções;

While

- Repete um bloco de instrução **enquanto** a condição definida no seu cabeçalho for verdadeira.



Estrutura de repetição

Diferenças

For

Algoritmo que recebe cinco números inteiros e exibe a soma dos números informados

While

Algoritmo que recebe números positivos e faça a soma dos número informados. Enquanto números positivos for informados, o algoritmo deve realizar a soma, senão o programa deve encerrar sua execução.

Enquanto

Sintaxe / algoritmo

**ENQUANTO (condição) FAÇA
INICIO**

// Lista de instruções

FIM

```
1 while(condição == True):  
2 |     #conjunto de instruções  
3
```

Estrutura de repetição

while

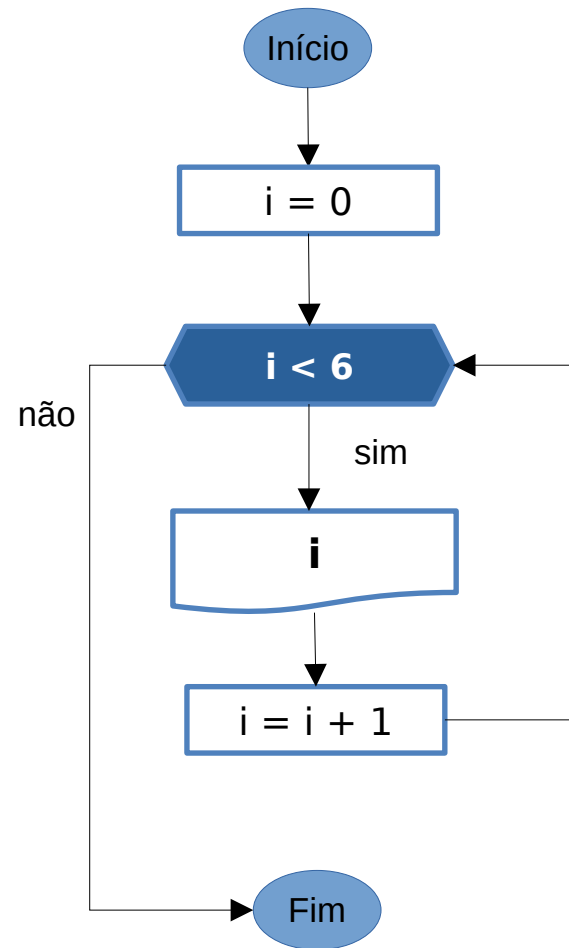
Exemplo

- Exibe o valor de **i enquanto** menor que 6

```
i = 1
```

```
while i < 6:  
    print(i)  
    i += 1
```

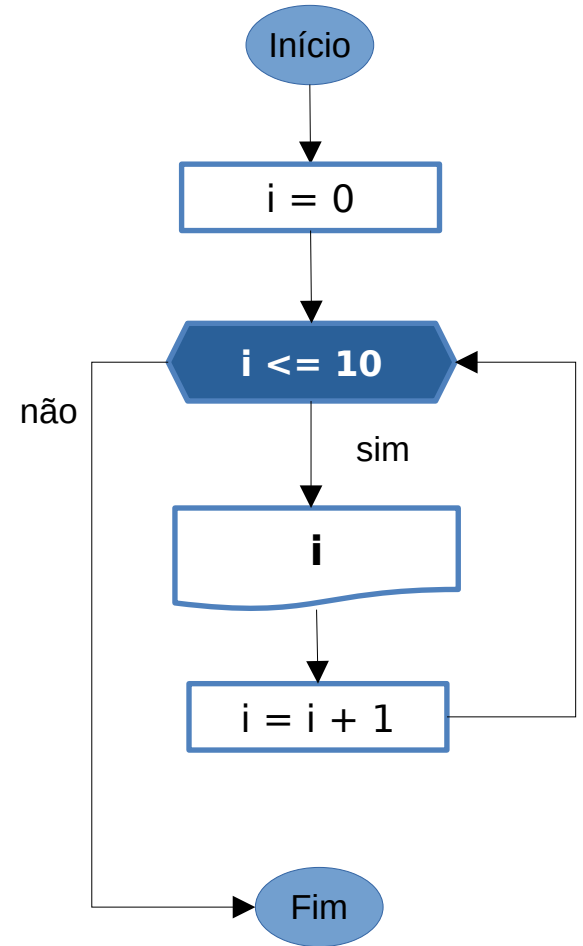
1
2
3
4
5



Nota: Lembrar de incrementar o i, ou então, o loop continuará para sempre:

Atividade 01

Faça um algoritmo que escreve os números de 1 até 10 (incluindo). Use a estrutura de repetição **while**.



Pseudocódigo / Fluxograma

ALGORITMO

$i \leftarrow 1$

Enquanto ($i \leq 10$) faça

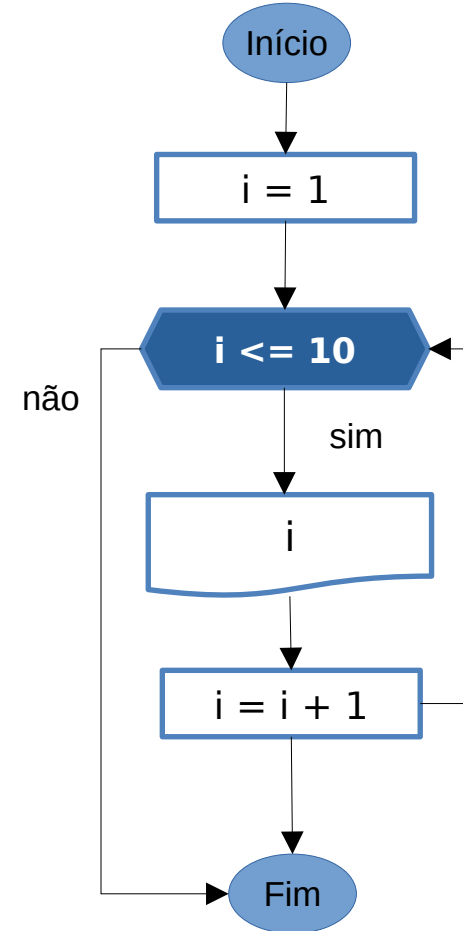
 Início

 Escreva i

$i \leftarrow i + 1$

 Fim

FIM_ALGORITMO



Algoritmo em Python

```
i = 1

while i <= 10:
    print(i)
    i = i + 1
```

ALGORITMO

$i \leftarrow 1$

Enquanto ($i \leq 10$) faça

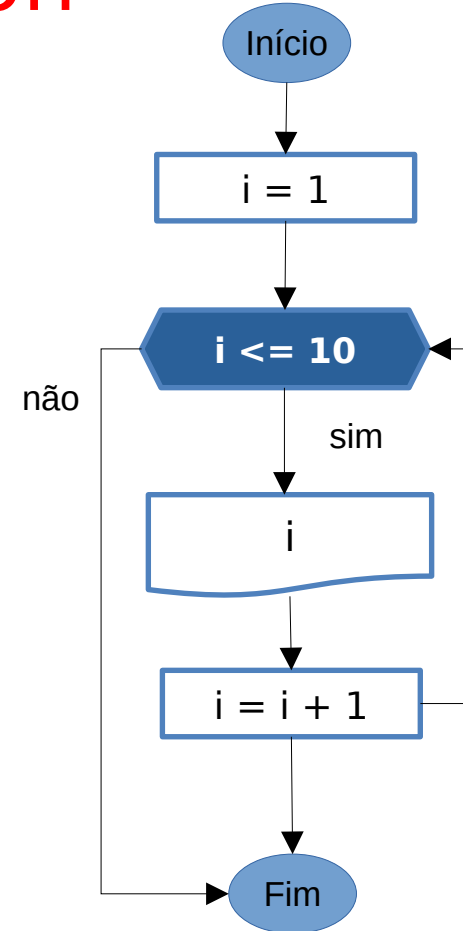
 Início

 Escreva i

$i \leftarrow i + 1$

 Fim

FIM_ALGORITMO



Atividade 02

Faça um fluxograma que leia números informados pelo usuário. O algoritmo deve escrever o dobro do número informado. **Enquanto** o usuário informar um número positivo, o programa deve solicitar novamente um novo número. **Quando um número negativo** for digitado, o algoritmo deverá parar de ler números.

Pseudocódigo / Fluxograma

Atividade 02

ALGORITMO

Escreva "digite um número:"

Leia **num**

Enquanto (**num** \geq 0) Faça

 Início

dobro \leftarrow **num** * 2

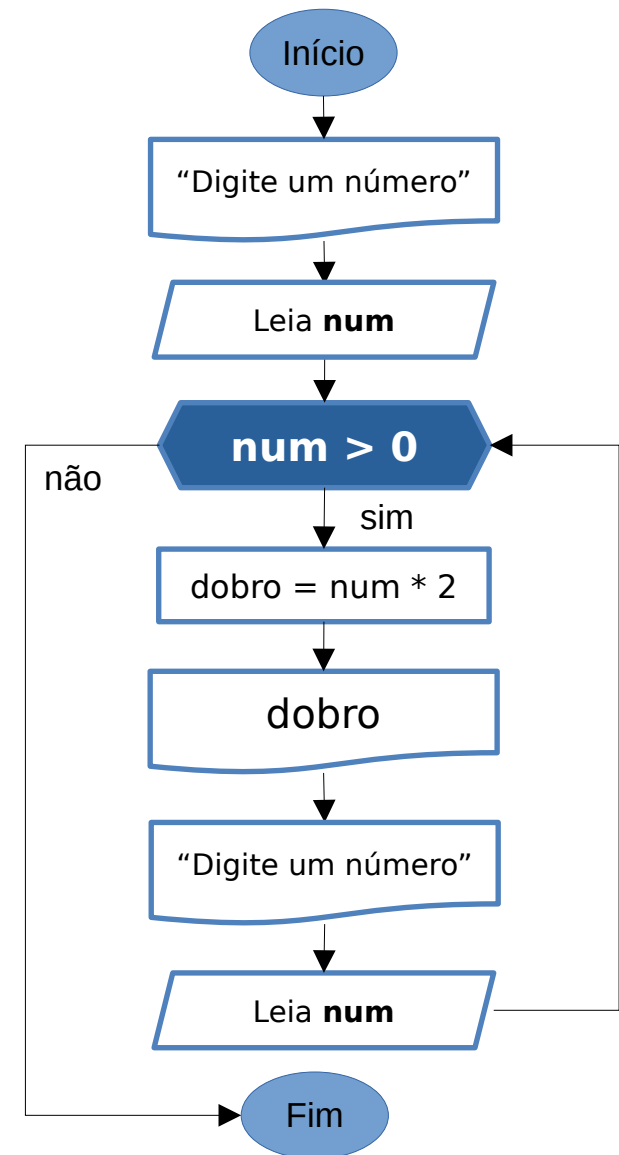
 Escreva **dobro**

 Escreva "Digite um
número:"

 Leia **num**

 Fim

FIM_ALGORITMO



Algoritmo

Atividade 02

```
num = int(input("Digite um número"))

while num > 0:
    dobro = num * 2
    print(dobro)
    num = int(input("Digite um número"))
```

ALGORITMO

Escreva "digite um número:"

Leia numero

Enquanto (numero >= 0) Faça

 Início

 dobro <- numero * 2

 Escreva dobro

 Escreva "Digite um número:"

 Leia numero

 Fim

FIM_ALGORITMO

Estruturas de Repetição

Enquanto ... faça

Atenção: Cuidado com **laços infinitos**! Devemos nos certificar que em algum momento a expressão lógica próxima a palavra reservada **enquanto** será **FALSA**!

Estrutura de Repetição

Exemplo: Algoritmo com laço infinito

```
ALGORITMO
```

```
  contador ← 0
```

```
  Enquanto (contador < 10) FACA
```

```
    Início
```

```
      contador -= 1
```

```
      Escreva contador
```

```
      contador += 1
```

```
    Fim
```

```
FIM_ALGORITMO
```

Exercícios

Exercício 01

Aula08_SeuNome.py

Escreva um algoritmo que recebe números inteiros do usuário. **Enquanto** o número positivo for digitado, o algoritmo deve verificar **se** o número informado é menor que 5, **senão** informe que o número é maior que 5. O programa deve encerrar se o número informado for negativo.

Pseudocódigo

ALGORITMO

Declare número Numérico

Escreva "Digite um número:"

Leia número

Enquanto (número \geq 0) faça

Início

Se (número $<$ 5) então

Início

Escreva "O número
digitado é menor que 5!"

Fim

Escreva "Digite um número:"

Leia número

Fim

FIM_ALGORITMO

Exercício 02

Aula08_SeuNome.py

Faça um algoritmo que lê a idade de pessoas. O programa deve solicitar a idade, **enquanto** a idade informada pelo usuário estiver entre 0 e 120. O algoritmo deve somar e exibir a quantidade de pessoas informadas em cada faixa etária. Considere as faixas etárias apresentadas abaixo:

FAIXA ETÁRIA

Menor

Adulto

Idoso

IDADE

≤ 18 anos

de 19 a 60 anos

> 60 anos

Pseudocódigo

ALGORITMO

Declare *idade*, *faixa1*, *faixa2*, *faixa3* Numérico

Escreva "Digite sua idade: "

Leia *idade*

Enquanto (*idade* \geq 0 e *idade* \leq 120) Faça

Início

Se (*idade* \leq 18) Então

Início

faixa1 \leftarrow *faixa1* + 1

Fim

Continuação

Se (idade \geq 18) e (idade \leq 60) Então

Início

faixa2 \leftarrow faixa2 + 1

Fim

Se (idade > 60) Então

Início

faixa3 \leftarrow faixa3 + 1

Fim

Escreva "Digite sua idade:"

Leia idade

Fim

Escreva "Quantidade de pessoas na faixa 1: ", faixa1

Escreva "Quantidade de pessoas na faixa 2: ", faixa2

Escreva "Quantidade de pessoas na faixa 3: ", faixa3

FIM_ALGORITMO

Exercício 03

Aula08_SeuNome.py

Faça um algoritmo que recebe números inteiros. **Enquanto** o número -100 não for digitado, o programa deve solicitar para o usuário digitar um novo número. **Quando** o usuário informar o valor -100, o programa deve exibir o maior número entre todos informados pelo usuário;

Pseudocódigo

ALGORITMO

Declare número, maior Numérico

Escreva “digite um número:”

Leia número

Enquanto (número != -100) Faça

Início

Se (número > maior) Então

Início

maior <- número

Fim

Escreva “Digite outro número:”

Leia número

Fim

Escreva “O maior número digitado: ”, maior

FIM_ALGORITMO

Exercício 04

Aula08_SeuNome.py

Faça um programa que leia números reais **enquanto** o número zero não for digitado. Quando for inserido o número zero, o programa deverá apresentar quantos números foram inseridos e a média dos números informados.

Exercício 05

Aula08Ex05.py

Escreva um algoritmo que recebe um número inteiro. O programa deve mostrar os números pares e ímpares (separados, em duas colunas), de 1 até o número inteiro informado.

Exercício 06

Aula08_SeuNome.py

Faça um algoritmo que solicite ao usuário um número entre 12 e 20. Se o usuário digitar um número que não está entre esse intervalo, o algoritmo deve mostrar a mensagem "entrada inválida" e solicitar o número novamente. Se digitar correto mostrar o número digitado.

Exercício 07

Aula08_SeuNome.py

Leia um número positivo do usuário. Calcule e imprima a sequência Fibonacci até o primeiro número superior ao número lido.

0, 1, 1, 2, 3, 5, 8, 13, 21, 34..

Entrada
30

Saída
0
1
1
...
13
21

Exercício 08

Aula08_SeuNome.py

Escreva um algoritmo que recebe números inteiros do usuário. O programa deve solicitar números do usuário **enquanto** o valor informado seja ímpar. Some todos os números ímpar e faça a subtração com o número par.

Entrada

1
3
5
7
9
10



execução

4
9
16
23

Saída

Diferença: 13

Exercício 09

Aula08_SeuNome.py

Escreva um algoritmo que recebe números inteiros do usuário. O programa deve receber números do usuário, **enquanto** a soma de todos os números informados não seja divisível por 7.

Entrada

1
8
10
2



execução

1
9
19
21

Saída

21 é divisível por 7.

Exercício 10

Aula08_SeuNome.py

Elabora você mesmo um exercício e resolva.
Precisa envolver a estrutura **ENQUANTO**.

Próxima aula

Estrutura de Repetição

- Do while