

# Estruturas de Controle

- ESTRUTURA SEQUENCIAL
- ESTRUTURAS CONDICIONAIS
  - Estrutura Condicional Simples
  - Estrutura Condicional Composta
  - Seleção entre duas ou mais Sequências de Comandos
- **ESTRUTURA DE REPETIÇÃO**
  - Estrutura de Repetição em Algoritmos
  - Estrutura de Repetição em Linguagem C
  - Comando de controle de laço

# Estrutura de Repetição

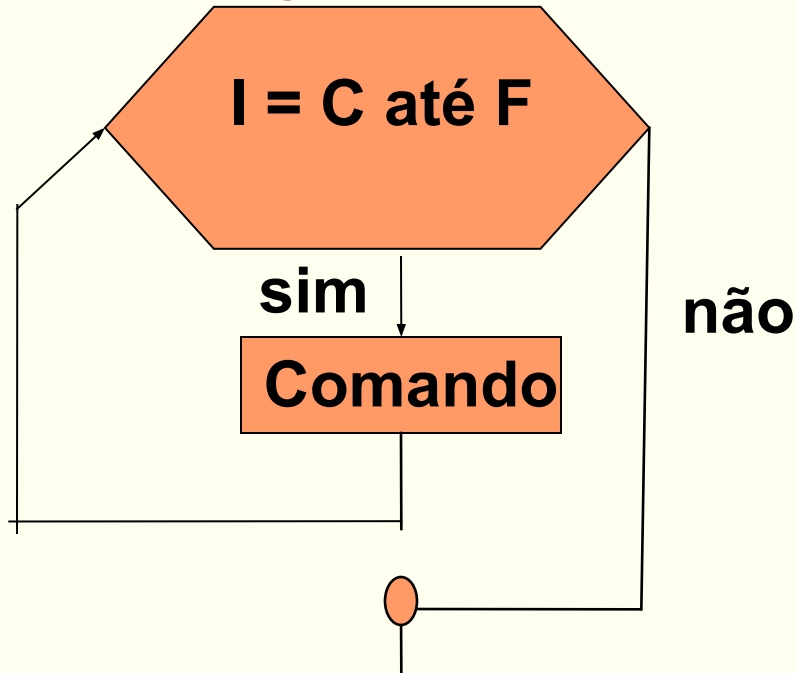
- Uma estrutura de repetição é utilizada quando um comando ou um bloco de comandos deve ser repetido.
- A quantidade de repetições pode ser fixa ou pode depender de uma determinada condição.
- O teste da condição pode ocorrer no início ou no final da estrutura de repetição.

# Estrutura de Repetição em Algoritmo

- Três tipos de estruturas de repetição serão consideradas na elaboração de Algoritmos:
  - ✓ Repetição Contada
  - ✓ Repetição com teste no início
  - ✓ Repetição com teste no final

# Repetição Contada

## Fluxograma



## Linguagem Python

```
for I in range(C,F):  
    comando
```

# Comando for

```
for <variável-contador> in  
range(<valores_possíveis>):  
    instrução
```

```
for <variável-contador> in  
range(<valores_possíveis>):
```

```
    instrução_1
```

```
    instrução_2
```

```
    ...
```

```
    instrução_n
```

# Comando for

`range(início,fim,passo)`

início: primeiro elemento do conjunto de valores

default: início = 0

fim: último elemento do conjunto de valores

passo: tamanho do incremento ou decremento

default: passo = 1

A repetição persiste até `contador = fim - 1`

# Exemplos 1 - 8

## Códigos

# Repetição Contada

## Exemplo 9

Dado um conjunto de  $N$  números, calcular a média aritmética dos mesmos



# Repetição Contada

## Exemplo 10

Ler um número inteiro  $\geq 0$   
e calcular seu fatorial.

# Exercícios

1. Faça um programa que recebe um valor inteiro n e exibe os números pares a partir de n. Exemplo:
  - N = 10
  - **Saídas:** 0, 2, 4, 6, 8, 10.
  
2. Construa um código que apresente os valores inteiros maiores que 10 e menores que 25.
  
3. Faça um programa que recebe um valor inteiro n e exiba os números, a partir de n, de trás para frente e um intervalo de 3 em 3, Exemplo:
  - N = 10
  - **Saídas:** 10 7 4 1

# Exercícios

4. Faça um programa que leia um valor N inteiro e positivo, calcule e mostre o valor de E conforme a fórmula a seguir:

$$E = 1 + 1/1! + 1/2! + 1/3! + \dots + 1/N!$$

5. Faça um programa que receba números inteiros maiores que 1. A cada valor recebido, exiba um mensagem dizendo se o número é primo ou não.

# Exercícios

6. Faça um programa que receba valores inteiros positivos e calcule e mostre:

A soma dos números digitados.

A quantidade de números digitados.

A média dos números digitados.

O maior número digitado.

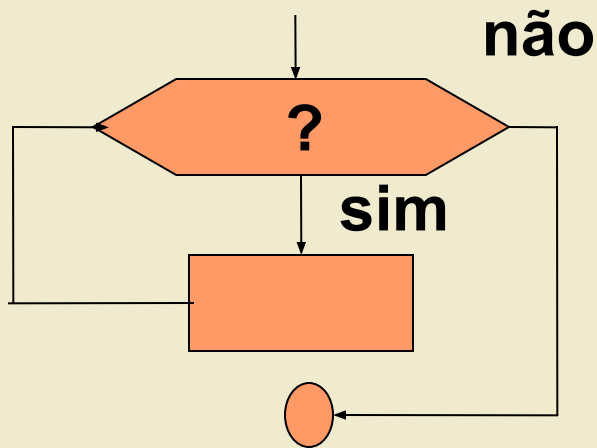
O menor número digitado.

A média dos números pares.

A porcentagem dos números ímpares entre todos os números digitados.

# Repetição com Teste no Início

Fluxograma



Linguagem Python

`while` (condição):  
comando

# Comando while

```
while (expressão de teste):  
    instrução
```

```
while (expressão de teste):  
    instrução_1  
    instrução_2  
    ...  
    instrução_n
```

# Comando while

- “while” pode substituir o comando “for” da seguinte maneira:

Inicialização da variável de teste - início

while(teste-**fim**):

    incremento da variável de teste - **passo**

# Comando while

- Há equivalência entre comandos for e while

```
for i in range(a,b,passo):  
    instrução
```

```
i = a  
while(i!=b):  
    instrução1;  
    i+=1
```

#a e b números



# Exemplos 11 e 12

# Repetição com Teste no Início

## Exemplo 13

```
sexo=input("sexo=")
```

```
while(sexo!="F")and(sexo!="f")and(sexo!="M")and(sexo!="m"):
```

```
    print("ERRO")
```

```
    sexo=input("sexo=")
```

```
print(sexo)
```

# Repetição com Teste no Início

Esta variável de controle deve ter um valor conhecido. Neste caso ela foi lida antes.

```
sexo=input("sexo=")
```

```
while(sexo!="F")and(sexo!="f")and(sexo!="M")and(sexo!="m"):
```

```
    print("ERRO")
```

```
    sexo=input("sexo=")
```

```
print(sexo)
```

A variável de controle deve ter seu valor modificado dentro do "laço". Neste caso a variável foi lida novamente.

laço

# Repetição com Teste no Início

## Exemplo 14

Ler  $N$  números reais (o valor de  $N$  também deve ser lido), sendo impressa a média desses números.

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

```
i=0
```

```
media = 0
```

```
while(i<n):
```

```
    valor=int(input("valor="))
```

```
    media += valor
```

```
    i+=1
```

```
print("média=",media/n)
```

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

```
i=0
```

```
media = 0
```

```
while(i<n):
```

```
    valor=int(input("valor="))
```

```
    media += valor
```

```
    i+=1
```

```
print("média=",media/n)
```

i é um contador  
que determina a  
parada

i é inicializado

i é incrementado de 1

# Comando break

- Utilizado no corpo de qualquer estrutura de laço.
- Causa a saída imediata do laço, desviando o programa para a próxima instrução após o laço atual.
- Se estiver em laços aninhados, o **break** afetará somente o laço que o contém e seus laços internos.
- No caso de laços como **for** e **while**, o laço é interrompido e os comandos do programa são retomados a partir da primeira linha fora do laço.

# Comando break

## Exemplo 15:

```
import math
```

```
sum=0
```

```
while(1):
```

```
    x=int(input("x="))
```

```
    if(x<0):
```

```
        break
```

```
    sum += math.sqrt(x)
```

```
print("sum=",sum)
```



# Comando continue

- Força a execução da próxima iteração do laço, não executando o código que vem a seguir.
- Esse comando ocorre apenas nos comandos “for” e “while”.

# Comando continue

- No caso do “while” e “do-while” a execução é desviada para o teste condicional e depois segue para o corpo do laço.
  - Interrompe a sequência de execuções dentro do laço,
  - Verifica a condição.
  
- No caso do “for”, o desvio é feito para o incremento, seguido pelo teste condicional e corpo do laço.
  - Interrompe a sequência de execuções dentro do laço,
  - Incrementa ou decrementa o contador,
  - Verifica a condição.

# Comando continue

## Exemplo:

```
for valor in range(1,10):  
    if(valor%2):  
        continue  
    print(valor,end="")
```

# Comando Continue

- O incremento (atualização do índice) ocorre após o continue em um comando for.
- O incremento (atualização do índice) não ocorre no comando while, se o comando continue vier antes do incremento.

## Exemplo:

```
valor = 1
while(valor < 10):
    if(valor%2):
        continue
    print(" ",valor,end="")
    valor+=1
```