



Estruturas de repetição – comando for

ICC - Introdução à Ciência da Computação

UFCG/UASC

Prof. Roberto M. de Faria

Estruturas de repetição

Conteúdo:

- Estruturas de repetição
- O comando **for**
- Exemplos com o uso do comando **for**
- Comando **for** com estruturas condicionais
- Comandos **for** aninhados
- Exercícios de fixação

Estruturas de repetição

- **Estrutura Sequencial:** *conjunto de instruções executadas em sequência linear, de cima para baixo;*
- **Estrutura Condicional:** *conjunto de instruções que são ou não executados dependendo de condições explícitas;*
- **Estrutura de Repetição:** *conjunto de instruções que são executados uma ou repetidas vezes dependendo de condições explícitas.*

Estruturas de repetição

- E em Python?
- Qual comando/estrutura podemos usar para **montar estruturas de repetição** em nossos algoritmos?



Estruturas de repetição

- E em Python?
- Qual comando/estrutura podemos usar para **montar estruturas de repetição** em nossos algoritmos?
- Estruturas de Repetição
 - for
 - for aninhado
 - while
 - while aninhado
 - while/for aninhado

Estruturas de repetição

Definição:

Loop, Laço ou **Laçada** é: trecho de programa executado repetidamente

- Pode ser executado em um *número fixo de repetições*;
- Pode ser executado *enquanto uma condição* seja satisfeita;
- Pode ser executado *até que uma condição* seja satisfeita.

Estruturas de repetição



Exemplo:

- Como faríamos para escrever na tela todos os números pares entre 0 e 100 (inclusive)?
 - Solução 1: Escrever vários (**MUITOS!**) comandos **print** compreendendo todo o intervalo

Estruturas de repetição

Exemplo (Solução 1):

```
if 0%2==0:
    print(0)
if 1%2==0:
    print(1)
if 2%2==0:
    print(2)
if 3%2==0:
    print(3)
if 4%2==0:
    print(4)
if 5%2==0:
    print(5)
...
if 100%2==0:
    print(100)
```



Estruturas de repetição

Exemplo (Solução 1):

```
if 0%2==0:
    print(0)
if 1%2==0:
    print(1)
if 2%2==0:
    print(2)
if 3%2==0:
    print(3)
if 4%2==0:
    print(4)
if 5%2==0:
    print(5)
...
if 100%2==0:
    print(100)
```

Tá ficando legal!

Vai acabar
rapidinho!
#sqn



Estruturas de repetição

Exemplo (Solução 1):

```
if 0%2==0:
    print(0)
if 1%2==0:
    print(1)
if 2%2==0:
    print(2)
if 3%2==0:
    print(3)
if 4%2==0:
    print(4)
if 5%2==0:
    print(5)
...
if 100%2==0:
    print(100)
```

Tem uma maneira MUITO melhor!



Estruturas de repetição

Exemplo:

- Como faríamos para escrever na tela todos os números pares entre 0 e 100, inclusive?
 - Solução 1: Escrever vários (**MUITOS!**) comandos **print** compreendendo todo o intervalo
 - Solução 2: Utilizando um *comando de repetição, laço ou loop*

Estruturas de repetição

Estruturas de Repetição: permitem que uma ação seja executada mais de uma vez **sem que tenhamos que escrever** o comando ou conjunto de comandos **várias vezes**.

Ponto fundamental: identifique o **padrão de comportamento**

- **O que repete** várias vezes?
- **O que muda** em cada repetição?
- **Quais as condições** para repetir?
- **Quantas vezes** deve repetir?

O comando **for**

- Um comando de repetição executa várias vezes um comando ou conjunto de comandos, sem que seja necessário reescrevê-los;
 - Python possui dois comandos de repetição: o **for** e o **while**;
 - O laço **for** tem o mesmo comportamento do laço **while**: REPETIR a execução de um comando ou conjunto de comandos;
 - A principal **diferença** entre os laços está na forma de estabelecer as condições para repetição e parada;
- ✓ **Dica** : usa-se um comando **for** quando há uma **definição previsível e finita do número de repetições**.

O comando **for**

Em Python:

Um contador de repetições

```
...  
for i in range(início, fim, passo) :  
    comando1  
    comando2  
    comando3  
    ...  
    comandoN  
...
```

O comando for

Em Python:

Um contador de repetições

```
...  
for i in range (início, fim, passo) :  
    comando1  
    comando2  
    comando3  
    ...  
    comandoN  
...
```

O valor inicial
do contador
(inicialização)

O comando for

Em Python:

Um contador de repetições

```
...  
for i in range(início, fim, passo) :  
    comando1  
    comando2  
    comando3  
    ...  
    comandoN  
...
```

O valor final do
contador
(*condição de parada*)

O comando for

Em Python:

Um contador de repetições

...

```
for i in range(inicio, fim, passo):
```

```
    comando1
```

```
    comando2
```

```
    comando3
```

```
    ...
```

```
    comandoN
```

...

O valor de
incremento/decremento
do contador a cada
repetição
(*atualização da condição
de parada*)

O comando **for**

Em Python:

...

```
for i in range(início, fim, passo) :
```

comando1

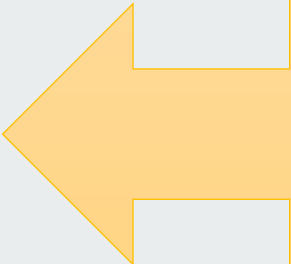
comando2

comando3

...

comandoN

...



Comandos a
serem executados
repetidamente
enquanto a
condição de
parada é **FALSA**

O comando for

Em Python:

```
...  
for i in range(início, fim, passo) :  
    comando1  
    comando2  
    comando3  
    ...  
    comandoN
```

|
...
|

Comandos estarão fora do laço de repetição quando **a indentação for recuada**

Exemplos com o uso do for

Exemplo 1: Exibir todos os números inteiros de 0 a 10.

```
for i in range(0,11,1):  
    print(i)
```

```
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
10
```

IMPORTANTE: O 11 é a **condição de parada** do laço. Logo, ele não é exibido pois a repetição é encerrada assim que **$i == 11$** .

Exemplos com o uso do **for**

Exemplo 2: Exibir todos os números pares entre 0 e 100 (inclusive).

```
for i in range(0,101,2):  
    print(i)
```

Exemplos com o uso do for

Exemplo 2: Exibir todos os números pares entre 0 e 100 (inclusive).

```
for i in range(0, 101, 2):  
    print(i)
```

**E se eu não quisesse
mostrar o 0 e o 100?**

Exemplos com o uso do **for**

Exemplo 2b: Exibir todos os números pares entre 0 e 100 (exclusive).

```
for i in range(2,100,2):  
    print(i)
```

Exemplos com o uso do for

Exemplo 3: Exibir todos os números entre 0 e 100 (inclusive), exceto os números 31 e 77.

```
for i in range(0,101,1):  
    if i==31 or i==77:  
        continue  
    print(i)
```


Exemplos com o uso do for

Exemplo 3: Exibir todos os números entre 0 e 100 (inclusive), exceto os números 31 e 77.

```
for i in range(0,101,1):  
    if i==31 or i==77:  
        continue  
    print(i)
```

Comando especial:

continue

Força o avanço do laço no ponto em que é inserido

Exemplos com o uso do **for**

Exemplo 3: Exibir todos os números entre 0 e 100 (inclusive), exceto os números 31 e 77.

```
for i in range(0,101,1):  
    if i==31 or i==77:  
        continue  
    print(i)
```

Essa linha de código não será executada toda vez que o comando **continue** for alcançado.

Exemplos com o uso do **for**

Exemplo 4: Exibir todos os números ímpares entre 0 e 100.

```
for i in range(1,100,2):  
    print(i)
```

Exemplos com o uso do for

Exemplo 5a: Exibir a soma de todos os números pares entre 0 e 100 (inclusive).

```
soma = 0
for i in range(0,101,2):
    soma = soma + i
print(soma)
```

2550

Exemplos com o uso do for

Exemplo 5b: Exibir a soma de todos os números pares entre 0 e 100 (exclusive).

```
soma = 0
for i in range(2,100,2):
    soma = soma + i
print(soma)
```

2450

Comando **for** com estruturas condicionais

- Programa que recebe um número n , fornecido pelo usuário, e mostra se n é ou não um número primo:

```
n = int(input("Digite o valor de n: "))
contador = 0                                # Contador de divisores
for i in range(2,n,1):
    if n%i == 0:                             # i é divisor de n ?
        contador = contador + 1
        break                               # Se já encontrou um divisor, pra que
                                           # continuar testando?
if n != 1 and contador == 0:
    print("%d é primo!" % n)
else:
    print("%d não é primo! " % n)
```

Comando **for** com estruturas condicionais

- Programa que recebe um número positivo n , de 5 dígitos e mostra a quantidade de dígitos pares deste número:

```
n = int(input("Digite o valor de n, com 5 dígitos: "))
contador = 0
for i in range(1, 6, 1):
    digito = n % 10                # Extrai o dígito da direita
    if digito%2 == 0:              # O dígito é par?
        contador = contador + 1  # Conte se o dígito é par
    n = n // 10                   # Elimina o dígito da direita
print("O número possui %d dígitos pares!" % contador)
```

Comandos **for** aninhados

- Programa que recebe uma quantidade de linhas e desenha um triângulo de asterisco como no modelo a seguir, para 4 linhas:

```
    *
   **
  ***
 ****

num_lin = int(input("Informe o número de linhas: "))
contador = 0
for linha in range(1, num_lin+1, 1):
    for coluna in range(0, num_lin-linha, 1):
        print(" ", end="")
    for coluna in range(1, linha+1, 1):
        print("*", end="")
    print()
```


Comandos **for** aninhados

- Programa que dados os valores inteiros de n e m , calcula o valor de x pela fórmula e seguir:

$$x = \sum_{i=1}^n \sum_{j=1}^m (i + j)$$

```
n = int(input("Informe o valor de n: "))
m = int(input("Informe o valor de m: "))
x = 0
for i in range(1, n+1, 1):
    for j in range(1, m+1, 1):
        x = x + (i + j)
print("x = %d" % x)
```

Exercícios de fixação

1. Faça um programa que calcule e imprima o IMC de N pessoas.
2. Faça um programa que calcule e mostre o fatorial de um numero natural.
3. Faça um programa que some os números naturais no intervalo de M a N, inclusive. M pode ser menor, maior ou igual a N.
4. Faça um programa que calcule e mostre o produto de N números inteiros positivos quaisquer.

Estruturas de repetição

Conteúdo:

- Estruturas de repetição
- O comando **for**
- Exemplos com o uso do comando **for**
- Comando **for** com estruturas condicionais
- Comandos **for** aninhados
- Exercícios de fixação

Referências e créditos

- Estes slides foram compilados a partir dos trabalhos:
 - do Prof. Maxwell Guimarães na disciplina Introdução à Programação/UASC/UFCG
 - do Prof. Roberto Faria nas disciplinas Introdução à Ciência da Computação e Introdução à Programação/UASC/UFCG