

Programação de Computadores

Aula 10

Profa. Cristiane

Estruturas de repetição

- Também conhecidas como laços (loop);
- Utilizadas para executar repetidamente uma instrução ou um bloco de instruções enquanto determinada condição for verdadeira.
- São três:

Pseudocódigo	Python
para	for
enquanto	while
faça...enquanto	-

Estruturas de repetição

- Para determinarmos qual é a estrutura mais adequada, devemos saber:
 - o número de vezes que o trecho programa vai ser executado → **laços contados**
 - ou a condição para que ela aconteça → **laços condicionais**
- **Laços contados:** auxílio de um **contador**, que auxiliará no laço. Este possibilita a repetição da estrutura até que o contador atinja o limite estipulado na condição.
- **Laços condicionais:** o valor é desconhecido e devemos utilizar uma variável com valor predefinido em uma condição dentro do laço para finalizarmos a repetição

Estruturas de repetição

- Independente do tipo de laço, todos são constituídos de três partes:
 - Inicialização(ões) da(s) variável(is) de controle
 - Condição(ões)
 - Atualização da(s) variável(is) de controle

Estrutura para (for)

- É compacta, pois a inicialização, condição e atualização estão reunidos na declaração do laço.
- Algoritmo:

```
para (inicialização; condição; atualização)
{
    instruções
}
```

Comando para repetições na linguagem Python

for variável **in** lista:

comandos

continue (opcional)

break (opcional)



Repete os comandos para os valores de cada elemento na lista.

A cláusula **continue** pode ser usada para ignorar os comandos e executar a próxima iteração ou passo do laço mais interno.

A cláusula **break** permite sair do ciclo mais interno a qualquer momento.

O comando for na linguagem Python

Diferente de outras linguagens de programação, o comando for de Python itera sobre os itens de uma sequência (uma lista ou uma string), na ordem em que aparecem na sequência. Veja a saída deste programa:

```
# Exemplos simples do comando for
animais = ['gato', 'cachorro', 'rinoceronte', 'leão']
for a in animais:
    print(a, 'string de tamanho:', len(a))

sequencia = [0, 1, 2, 3, 4, 5]
for num in sequencia:
    print(num)
```

```
gato string de tamanho: 4
cachorro string de tamanho: 8
rinoceronte string de tamanho: 11
leão string de tamanho: 4
0
1
2
3
4
5
```

O comando for na linguagem Python - função range

A função range é útil para iterar sobre sequências numéricas, porque gera progressões aritméticas.

Exemplo	Saída
<pre>for i in range(10): print(i, end=" ")</pre>	0 1 2 3 4 5 6 7 8 9
<pre>for j in range(3, 8): print(j, end=" ")</pre>	3 4 5 6 7
<pre>for n in range(0, 21, 2): print(n, end=" ")</pre>	0 2 4 6 8 10 12 14 16 18 20
<pre>for k in range(5, 20, 3): print(k, end=" ")</pre>	5 8 11 14 17

Algoritmo Tabuada

Escreva um algoritmo que solicite ao usuário um número e calcula e mostra a tabuada desse número.

Algoritmo

Início

```
inteiro numero
escreva("Entre com o número:")
leia(numero)
```

```
escreva ( numero* 0 )
escreva ( numero* 1 )
escreva ( numero* 2 )
escreva ( numero* 3 )
escreva ( numero* 4 )
escreva ( numero* 5 )
escreva ( numero* 6 )
escreva ( numero* 7 )
escreva ( numero* 8 )
escreva ( numero* 9 )
escreva ( numero*10 )
```

fim

Algoritmo Tabuada com repetição

Algoritmo Início

```
inteiro numero, i
escreva("Entre com o número:")
leia(numero)
para (i=0; i<=10; i++){
    escreva ( numero*i )
}
fim
```

Teste de mesa:
Por exemplo: numero = 3

i	Saída
0	0
1	3
2	6
3	9
4	12
5	15
6	18
7	21
8	24
9	27
10	30

Exemplo

Faça um algoritmo que solicite ao usuário 10 números reais calcula e mostra a soma dos números digitados. Use a estrutura de repetição **for**

Algoritmo Somatório

Início

numerico n1, soma
soma=0

escreva("Digite um número: ")

leia(n1)

soma=soma+n1

Acumulador

1

2

3

4

5

escreva("Digite um número: ")

leia(n1)

soma=soma+n1

escreva("A somatória é: " + soma)

fim

6

7

8

9

10

Exemplo

Algoritmo Somatório início

```

    numérico n1, soma, i
    soma=0
    para(i=1;i<=10;i++) {
        escreva("Digite um número: ")
        leia(n1)
        soma=soma+n1
    }
    escreva("A somatória é: " + soma)
fim
  
```

Diagrama de fluxo:

- Acumulador:** soma
- Contador:** i

O bloco de código que contém a estrutura de repetição (loop) é destacado com um retângulo vermelho.

i	n1	soma
		0.0
1	5.5	5.5
2	1.5	7.0
3	2.0	9.0
4	3.5	12.5
5	7.5	20.0
6	2.5	22.5
7	1.0	23.5
8	2.0	25.5
9	4.5	40.0
10	12.5	52.5

Estrutura enquanto (*while*)

- Estrutura utilizada tanto para **laços contados** quanto para os **laços condicionais**, possui e a seguinte sintaxe:
- Algoritmo:

```
iniciar a variável de controle
enquanto (condição for verdadeira)
{
    instruções
    atualizar a variável de controle
}
```

Instrução que modifica o estado de algum elemento utilizado na condição

Comandos para repetições na linguagem Python

```
while condição:  
    comandos  
    continue (opcional)  
    break (opcional)
```



Repete os comandos enquanto a condição for verdadeira.

O comando **continue** pode ser usada para ignorar os comandos e executar a próxima iteração ou passo do laço mais interno.

O camando **break** permite sair do ciclo mais interno a qualquer momento.

Exemplo

Faça um algoritmo que calcula e mostra a média entre duas notas de **10 alunos.** Use a estrutura de repetição **enquanto**

Algoritmo exemploWhile

 numerico nota1, nota2, media, cont

 início

 cont = 1

enquanto (cont<=10) {

 escreva("Digite a primeira nota")

 leia(nota1)

 escreva("Digite a segunda nota")

 leia(nota2)

 media = (nota1+nota2)/2

 escreva("A média é " + media)

 cont=cont+1

 }

 Fim

Exemplo

Faça um algoritmo que calcula e mostra a média de uma quantidade indeterminada de números inteiros digitados pelo usuário. Use a estrutura de repetição **enquanto**.

Algoritmo exemploWhile2

```
numerico media, soma, num, qtde
caracter resp
```

Inicio

```
soma=0
```

```
qtde = 0
```

```
resp = 's'
```

```
enquanto (resp=='s" ou resp == 'S') {
```

```
    escreva("Digite um número")
```

```
    leia(num)
```

```
    soma=soma+num ← acumulador
```

```
    qtde=qtde+1 ← contador
```

```
    escreva("Deseja continuar?")
```

```
    leia(resp)
```

```
}
```

```
media=soma/qtde
```

```
escreva("A média dos números digitados é: " + media)
```

fim

Exercícios

- 1) Faça um programa em Python que imprima os números pares entre 0 e 100

- 2) Faça um programa em Python que imprima os números de 1 a 50 de 1 em 1 e de 52 a 100 de 2 em 2.

- 3) Faça um programa em Python que leia um valor n, inteiro e positivo, calcule e mostre a seguinte soma:

$$S = 1 + 1/2 + 1/3 + 1/4 + \dots + 1/n$$



www.cruzeirodosul.edu.br