



INSTITUTO FEDERAL
São Paulo
Câmpus São Carlos

Estruturas de Repetição

ALG – Algoritmos e Programação

Aula 11

Curso Técnico em Informática para Internet – Integrado ao Ensino Médio



INSTITUTO FEDERAL
São Paulo
Câmpus São Carlos

Objetivos da aula

- Conhecer as estruturas de repetição em Python
- Aplicar estruturas de repetição visando a criação de programas

Relembrando: Estruturas de Repetição

- Permitem que uma sequência de comandos seja executada repetidamente até que uma determinada condição de interrupção seja satisfeita
- O trecho do algoritmo em repetição é chamado de laço (ou “*loop*”)
- As repetições devem ser sempre finitas
- Todo laço possui um critério de parada

Relembrando: Estruturas de Repetição

- As Estruturas de Repetição se dividem em dois tipos:
 - Repetição não contada: para um número indefinido de repetições (embora também possa ser usada quando se conhece o número de repetições);
 - Repetição contada: para um número definido de repetições.

* Existe também a estrutura “sempre”, porém não será abordada no contexto dessa aula

Repetição Não Contada em Python

- Definida pelo comando **while**
- A repetição dos comandos está associada a uma condição
- Enquanto a condição tiver **valor verdadeiro**, o bloco de comandos será executado.
- Quando a condição passa a ter **valor falso**, a repetição termina.
- Sintaxe:

while <condição>:
 <bloco de comandos>

Condição para
continuar
a repetição

Repetição Não Contada em Python

- Exemplo:
- Imprimindo os números de 1 a 20

```
x = 1
while x <= 20:
    print (x)
    x = x + 1
|
```

Atenção: a variável de controle, no caso x, deve ter um valor conhecido antes de ser avaliada na condição.

- Somando os números pares do 2 ao 10:

```
soma = 0
x = 2
while x <= 10:
    soma = soma + x
    x = x + 2
print ("Soma total = ", soma)
|
```

Atenção: é preciso garantir que a variável de controle terá o seu valor modificado dentro do “laço”, para não entrar em *loop* infinito!

Repetição Não Contada em Python

- Exemplo:

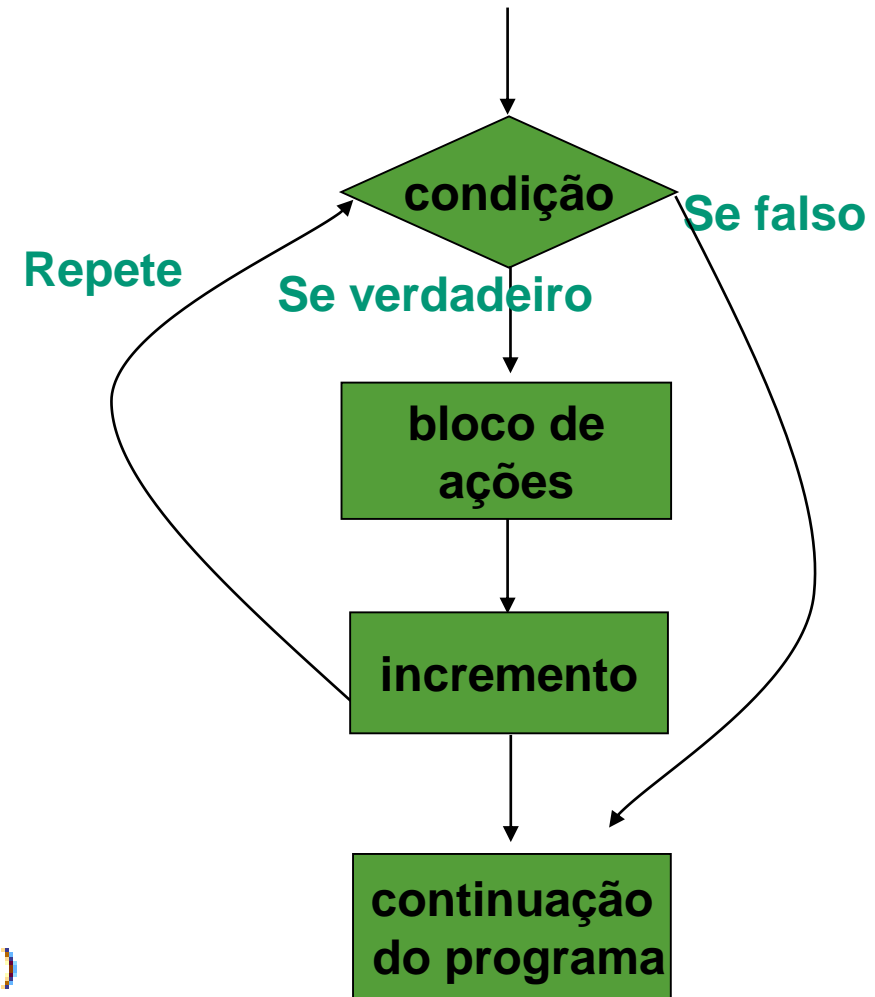
- Imprimindo os números de 1 a 20

```
x = 1
while x <= 20:
    print (x)
    x = x + 1
|
```

- Somando os números pares do 2 ao 10:

```
soma = 0
x = 2
while x <= 10:
    soma = soma + x
    x = x + 2
print ("Soma total = ", soma)
|
```

Fluxo de execução:



Estruturas de Repetição

- Dicas para elaborar um laço de repetição:
 - Definir o ponto inicial (valor inicial da variável de controle);
 - Definir o ponto de chegada (condição de parada);
 - Definir uma maneira de sair do ponto inicial e atingir a condição de parada.

Exercícios



Exercícios

1. Faça um programa que exiba todos os números de 1 a 100 que são divisíveis por 7.
2. Faça um programa que exiba todos os números de 1 a 100 que são divisíveis por 7 e por 3.
3. Faça um programa para mostrar a tabuada de um número qualquer fornecido pelo usuário. Por exemplo, se o número fornecido for igual a 3, o programa de apresentar a seguinte saída:

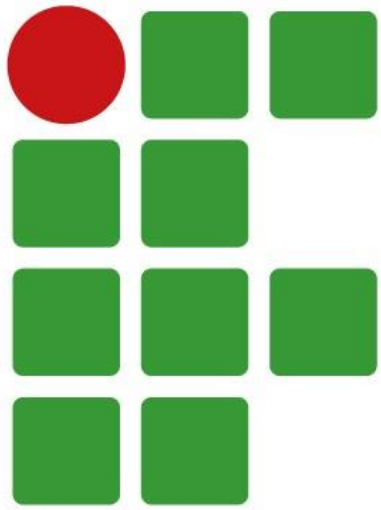
1 x 3 = 3
2 x 3 = 6
3 x 3 = 9
4 x 3 = 12
5 x 3 = 15
6 x 3 = 18
7 x 3 = 21
8 x 3 = 24
9 x 3 = 27

Exercícios (cont.)

4. Faça um programa que receba um número inteiro maior que 1, verifique se o número é primo ou não e mostre a mensagem de número primo ou de número não primo. Obs: Um número é primo quando é divisível apenas por 1 e por ele mesmo.
5. Faça um programa para calcular a área de um triângulo, que não permita a entrada de dados inválidos, ou seja, medidas menores ou iguais a zero. Obs: $\text{área do triângulo} = \text{base} * \text{altura} / 2$.
6. Faça um programa que receba dois números positivos, faça a divisão desses números e apresente o resultado. A execução do algoritmo será encerrada somente quando um dos números for negativo. Obs: não existe divisão por zero.

Exercícios (cont.)

7. Faça um programa para mostrar as tabuadas dos números de 1 a 10.
8. Faça um programa que leia um conjunto de valores correspondentes às notas que os alunos obtiveram em uma prova. Quando o valor fornecido for negativo, significa que não existem mais notas para serem lidas. Após a leitura das notas, escrever a média das notas da sala.
9. Faça um programa que mostre os 8 primeiros termos da sequência de Fibonacci. Ex: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55...
10. Faça um programa que leia um número inteiro ≥ 0 e calcule o seu fatorial.



INSTITUTO FEDERAL

São Paulo

Câmpus São Carlos