

INSTITUTO FEDERAL
BAIANO

Lógica e Linguagem de Programação

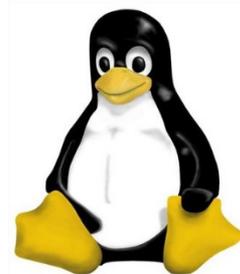
Python

Estruturas de Repetição

Prof. José Honorato Ferreira Nunes

honoratonunes@softwarelivre.org

<http://softwarelivre.org/zenorato>



Resumo da aula

Estruturas de Controle

Repetição

Pré-Teste (while)

Pós-Teste (do while)

Variável de controle (for)

Estruturas de Controle - Repetição

Existem três estruturas básicas para a construção de algoritmos, que são:

- algoritmos sequenciais,
- algoritmos com seleção;
- e algoritmos com repetição.

A combinação dessas três estruturas permite-nos a construção de algoritmos para a resolução de problemas extremamente complexos.

Estruturas de Controle - Repetição

- Nesta aula veremos as estruturas de repetição, possíveis em algoritmos e existentes na maioria das Linguagens de Programação.

Estruturas de Controle - Repetição

- Uma estrutura de repetição permite que uma sequência de instruções (comandos) seja executada várias vezes, até que uma condição (teste) seja satisfeita, ou seja, repete-se um conjunto de instruções sem que seja necessário escrevê-las várias vezes.
- As estruturas de repetição também são chamadas de Laços ou Loops.

Estruturas de Controle - Repetição

- Para sabermos quando utilizar uma estrutura de repetição, basta analisarmos se uma instrução ou uma sequência de instruções precisa ser executada várias vezes, se isto se confirmar, então deve-se utilizar uma estrutura de repetição.
- As estruturas de repetição, assim como a de decisão (seleção), envolvem a avaliação de uma condição (teste).

Repetição com pré-teste: while

Uma maneira possível de executar um laço é utilizando o comando while. Ele permite que o código fique sendo executado numa mesma parte do programa de acordo com uma determinada condição.

- o comando pode ser vazio, simples ou bloco;
- ele é executado desde que a condição seja verdadeira;
- testa a condição antes de executar o laço.

Repetição com Pré-Teste : while

A repetição com Pré-Teste é uma estrutura de looping que repete um bloco de comandos enquanto a expressão avaliada for verdadeira.

Sintaxe:

```
while (condição):  
    comandos
```

Repetição com Pré-Teste : while

Ex:

```
resp='s'  
while (resp=='s'):  
    resp=raw_input('Deseja continuar S-sim N- Nao?')
```

Atividades: utilizar estrutura de repetição com Pré-Teste (while)

- Crie um algoritmo que leia um valor inteiro para X e escreva na tela X^3 . O algoritmo deve continuar pedindo o valor de X até que o usuário informe 0 (zero), então o programa encerra.
- Desenvolva um algoritmo capaz de apresentar na tela o fatorial de um número inteiro informado pelo usuário.
- Crie um algoritmo que solicite a idade de 10 alunos e depois mostre a idade do mais velho.

Atividades

- Crie um algoritmo que realize as seguintes atividades:
 - a) Pergunte a quantidade de alunos da turma.
 - b) Solicite ao usuário a matrícula de cada um dos X alunos.

Repetição com variável de controle: for

- ✓ O comando for é de alguma maneira encontrado em todas linguagens procedurais de programação.
- ✓ Em sua forma mais simples, a inicialização é um comando de atribuição que o compilador usa para estabelecer a variável de controle do loop.
- ✓ A condição é uma expressão de relação que testa a variável de controle do loop contra algum valor para determinar quando o loop terminará.
- ✓ O incremento define a maneira como a variável de controle do loop será alterada cada vez que o computador repetir o loop.

Repetição com variável de controle: for

Sintaxe:

```
for variável in seqüência:  
    # bloco de comandos
```

Exemplos:

```
for x in range (1, 100):  
    print x
```

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

Repetição com variável de controle: for

Ex 3:

```
texto=""
for x in range(1, 11):
    for y in range(1, 40):
        texto = texto + "-"
    texto = texto + "\n"
print texto
```

```
-----
-----
-----
-----
```

Bibliografia

- ❑ MANZANO, Wilson Y. Yamaturni-São Paulo-SP. **Lógica estruturada para programação de computadores**, Ed. Érica 1997 e 2001.
- ❑ MORAES, Celso Roberto. **Estruturas de Dados e Algoritmos**. Ed. Érica, São Paulo
- ❑ LOPES, Anita. **Introdução à programação**. Rio de Janeiro: Campus, 2002.
- ❑ BENEDUZZI, Humberto M. e METZ, João A. **Lógica e Linguagem de Programação – Introdução ao Desenvolvimento de Software (1ª edição)**. Editora do Livro Técnico, 2010.
- ❑ SEBESTA, Robert W. **Conceitos de linguagens de programação**. 9. ed. Porto Alegre: Bookman, 2003.
- ❑ CORMEN, Thomas H. **Algoritmos: teoria e prática**. Rio de Janeiro: Campus, 2002.
- ❑ ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. **Fundamentos da programação de computadores: algoritmos, Pascal e C/C++ e Java**. 2. ed. São Paulo: Pearson Prentice Hall, 2008.
- ❑ ZIVIANI, Nivio. **Projeto de algoritmos com implementações em Pascal e C**. São Paulo: Pioneira Thomson Learning.