

# Programação II

## Estrutura de Repetição WHILE

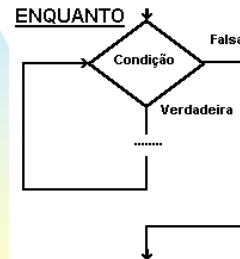
Jocélio Passos  
joceliopassos@bol.com.br

Ir p/ primeira página



## Estrutura de Repetição - ENQUANTO ou WHILE

- Repete enquanto a condição for verdadeira



06/02/2008

2

Ir p/ primeira página



## Estrutura de Repetição - WHILE

- Conhecido como laço controlado logicamente
- Por uma expressão booleana
- Não sabemos previamente quantas vezes o bloco será executado
- A condição é testada
  - Antes da primeira execução do bloco
  - E após cada repetição
- Pré-Teste X Pós-teste (DO WHILE)
- While com apenas um comando no corpo não precisa de chaves
- `while (x != 10) scanf ("%d", &x);`

06/02/2008

3

Ir p/ primeira página



## Estrutura de Repetição - WHILE

- Os mecanismos de controle :
  - continue ou loop (volta para a condição)
  - Exit ou Break (Termina o laço)
- São considerados parte da programação não-estruturada

06/02/2008

4

Ir p/ primeira página



## Estrutura de Repetição WHILE (Exemplos)

- Pseudo-código
  - FAÇA ENQUANTO (condição)
  - ...
  - FIMENQUANTO
- Pascal
  - WHILE (condição) DO
  - BEGIN
  - ...;
  - END;

06/02/2008

5

Ir p/ primeira página



## Estrutura de Repetição WHILE (Exemplos)

- Clipper
  - DO WHILE (condição)
  - ...
  - ENDDO
- C
  - while (condição)
  - {...;
  - };

06/02/2008

6

Ir p/ primeira página



## Estrutura de Repetição WHILE

- Exemplo : Pedir vários números (enquanto não for digitado zero) e mostrar o dobro de cada um.

```
◆ main ()
◆ {int num;
◆ printf ("\nDigite um numero : ");
◆ scanf ("%d",&num);
◆ while ( num != 0 )
◆ {printf ("\nDobro de %d = %d", num, num * 2);
◆ printf ("\nDigite outro numero : ");
◆ scanf ("%d",&num);}
```

06/02/2008

7

Ir p/ primeira página



## Estrutura de Repetição WHILE

- Algoritmo de leitura com While

```
◆ ...
◆ LEITURA;
◆ WHILE (CONDICAO)
◆ {...
◆ PROCESSAMENTO;
◆ ...
◆ LEITURA}
```

06/02/2008

8

Ir p/ primeira página



## Estrutura de Repetição WHILE

- Algoritmo de leitura com While

```
◆ ...
◆ LER A;
◆ WHILE (A != 0 )
◆ {LER B;
◆ LER C;
◆ PROCESSAMENTO;
◆ ...
◆ LER A;}
```

06/02/2008

9

Ir p/ primeira página



## Estrutura de Repetição WHILE

- Exemplo : Mostrar os 15 primeiros números

```
◆ {int i;
◆ i = 1;
◆ while ( i <= 15 )
◆ {printf ("%d, ", i);
◆ i++;}
◆ }
```

06/02/2008

10

Ir p/ primeira página



## Exercícios

- 1) Pedir e somar vários números enquanto não digitarem 0 (Zero).
  - a) Mostrar assim : "A soma dos números é 464"
  - b) Mostrar a soma parcial
  - c) Informar quantos números foram digitados
  - d) Média Aritmética
- 2) Pedir o nome de vários usuários e suas respectivas idades enquanto ele não digitar o nome "sair". Mostre se a pessoa é ou não adolescente (12 a 16 anos).
  - Obs : **while (strcmp (nome,"sair") != 0) { }**  
Equivala a : while (nome != "sair") do begin ... end;
- #include <string.h>

06/02/2008

11

Ir p/ primeira página



## Exercícios

- 3) Pedir vários anos, enquanto não for zero. Mostrar se cada ano é ou não bissexto.
  - No final mostrar quantos anos bissextos foram digitados.
  - Obs : O ano é bissexto quando for divisível por 4.
- 4) Determine se um número inteiro é ou não primo. Sair com zero. Um número é primo se é divisível apenas por um e por ele mesmo.

06/02/2008

### Leitura Recomendada

- ◆ **C Completo e Total, Capítulo 3, página 81 - 83**
- ◆ **Sobesta 298-302**

Ir p/ primeira página



### Exercícios Propostos

- 1) Na estrutura While o laço pode não ser executado nenhuma vez. Verdadeiro ou Falso ?
- 2) Pedir o sexo ('M' ou 'F') e a idade de várias pessoas e mostrar se a pessoa é adulta (maior que 21 anos). Deve sair quando for digitado 'T' na leitura do sexo.
- 3) Pedir e somar vários números enquanto não digitarem 0 (Zero).
  - ◆ a) somar só os pares
  - ◆ b) somar só os ímpares
  - ◆ c) Se o valor for par, somar, mas se for ímpar subtrair

06/02/2008

13

[Ir p/ primeira página](#) 

### Exercícios Propostos

- 4) Pedir um numero inteiro (enquanto não for digitado zero) e mostrar a soma dos seus dígitos.
- Exemplos:
  - a) Número = 1234. Resposta:  $4 + 3 + 2 + 1 = 10$
  - b) Número = 123. Resposta:  $3 + 2 + 1 = 6$
  - c) Número = 12. Resposta:  $2 + 1 = 3$
  - d) Número = 1. Resposta:  $1 = 1$

06/02/2008

14

[Ir p/ primeira página](#) 