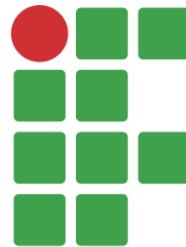


ESTRUTURA DE CONTROLE FOR DO/WHILE E SWITCH



**INSTITUTO
FEDERAL**
Paraíba

Professor: Paulo de Tarso F. Júnior
paulodt@gmail.com

Roteiro

- ▶ Estrutura de Repetição
- ▶ Exemplos de Uso da Estrutura For
- ▶ Estrutura de Seleção Múltipla Switch
- ▶ Estrutura de Repetição Do/While
- ▶ Instruções Break e Continue

Introdução

- ▶ Tópicos abordados
 - ▶ Estruturas adicionais de controle de repetição
 - ▶ For
 - ▶ Do/while
 - ▶ Estrutura de seleção múltipla **switch**
 - ▶ Instrução **break**
 - ▶ Usada para sair imediata e rapidamente de certas estruturas de controle
 - ▶ Instrução **continue**
 - ▶ Usada para saltar o restante do corpo de uma estrutura de repetição, procedendo com a próxima iteração do laço

Fundamentos

- ▶ Laço
 - ▶ Grupo de instruções que o computador executa repetidamente enquanto alguma condição permanece **verdadeira**
- ▶ Repetição controlada por contador
 - ▶ Repetição definida → Conhecimento do número de vezes de execução do laço
 - ▶ Uso de variável de controle → Contagem do número de repetições

Fundamentos

- ▶ Repetição controlada por sentinela
 - ▶ Repetição indefinida
 - ▶ Uso Falta de conhecimento prévio do número de repetições
 - ▶ Valor da sentinela → Término dos dados

Fundamentos

- ▶ Exigências da repetição controlada por contador
 - ▶ Nome de uma variável de controle (ou contador de laço)
 - ▶ Valor inicial da variável de controle
 - ▶ Condição de teste do valor final da variável de controle (i.e. se o laço deve continuar)
 - ▶ Incremento (ou decremento) para a alteração da variável de controle a execução do laço (iteração)

Fundamentos - While

► Exemplo

```
int contador = 1; // inicialização  
while (contador <= 10) { // condição de repetição  
    printf( "%d\n", contador );  
    +contador; // incremento  
}
```

► Declaração **int contador = 1;**

- Fornecimento do nome **contador** para a variável
- Declaração de um tipo (inteiro) para a variável
- Reserva de espaço para armazenamento da variável na memória
- Definição do valor inicial da variável como 1

Fundamentos - For

- ▶ Formato geral dos laços **for**

- ▶ *for (inicialização; teste_de_continuação_do_laço; incremento) // instrução (ões) a ser (em) executada (s)*

- ▶ Exemplo

- ```
for (contador = 1; contador <= 10; contador++)
 printf("%d\n", contador);
```

- ▶ Impressão dos inteiros de **1** a **10**

- Sem ponto-e-vírgula

- (;) após a última

- expressão



# Fundamentos - For/While

- Possibilidade de eventual substituição de laços **for** por laços **while**

```
inicialização;
while (teste_de_continuação_do_laço){
 instrução;
 incremento; (ou decremento;)
}
```

- Inicialização e incremento

- Possibilidade de uso de listas separadas por vírgulas
- Exemplo

```
for (i = 0, j = 0; j + i <= 10; j++, i++)
printf("%d\n", j + i);
```

# Fundamentos - For

- ▶ Notas sobre a estrutura **for**
  - ▶ Incremento pode ser negativo (decremento)
  - ▶ Se a condição de continuação do laço é inicialmente **falsa**
    - ▶ O corpo da estrutura **for** não é realizada
    - ▶ A execução prossegue com a próxima instrução após a estrutura **for**
  - ▶ Variável de Controle
    - ▶ Frequentemente (mas *não necessariamente*) *impressa ou usada dentro do corpo*

# Fundamentos - For

```
01 /* total com for */
02 #include <stdio.h>
03
04 int main()
05 {
06 /* inicializa total e declara numero*/
07 int total = 0, numero;
08 for (numero = 2; numero <= 100; numero += 2)
09 total += numero;
10
11 printf("A total eh %d\n", total);
12
13 return 0;
14 }
```

**A total eh 2550**

# Fundamentos - Switch

## ► Switch

- Uso em contextos de testes separados para cada um dos valores passíveis de serem assumidos por uma variável ou expressão → Decisões múltiplas

## ► Formato

- Série de rótulos case e um rótulo opcional default

```
switch (valor) {
 case '1':
 ações
 case '2':
 ações
 default:
 ações
}
```

- **break;** Abandono da estrutura

```
01 Contagem de conceitos */
02 #include <stdio.h>
03
04 int main()
05 {
06 char conceito;
07 int contaA = 0, contaB = 0, contaC = 0,
08 contaD = 0, contaF = 0;
09
10 printf("Entre com o conceito.\n");
11 scanf("%c",&conceito);
12
13 switch (conceito) { /* switch
14 case 'A': case 'a': /* conceito foi A maiúsculo ou minúsculo */
15 printf("Sua nota eh 10!"); /* instrção a ser executada*/
16 break;
17 case 'B': case 'b': /* conceito foi B maiúsculo ou minúsculo */
18 printf("Sua nota eh 9!"); /* instrção a ser executada*/
19 break;
20 case 'C': case 'c': /* conceito foi C maiúsculo ou minúsculo */
21 printf("Sua nota eh 8!"); /* instrção a ser executada*/
22 break;
23 case 'D': case 'd': /* conceito foi D maiúsculo ou minúsculo */
24 printf("Sua nota eh 7!"); /* instrção a ser executada*/
25 break;
```

# Fundamentos

```
29 default: /* demais conceitos */
30 printf("Fornecido um conceito incorreto.");
31 break;
32 }

33 printf("A: %d\n", contaA);
34 system ("pause");
35 }
```

# Fundamentos - Do/While

- ▶ A estrutura de repetição **do/while**
  - ▶ Similar à estrutura do **while**
  - ▶ Teste da condição para repetição → Após a execução do corpo do laço
    - ▶ Realização das ações → Pelo menos uma vez
- ▶ Formato

```
do {
 instrução;
} while (condição);
```

# Fundamentos - Do/While

## ► Exemplo

```
int contador = 1;
do {
 printf("%d ", contador);
} while (++contador <= 10);
```

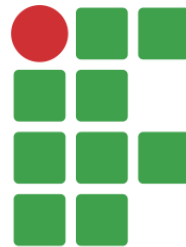
## ► Resultado Impressão dos inteiros de 1 a 10



# Fundamentos - Do/While

```
01 /* Uso da estrutura de repetição do/while */
02 #include <stdio.h>
03
04 int main()
05 {
06 int contador = 1;
07 do {
08 printf("%d ", contador);
09 } while (++contador <= 10);
10 return 0;
14 }
```

# ESTRUTURA DE CONTROLE FOR DO/WHILE E SWITCH



**INSTITUTO  
FEDERAL**  
Paraíba

Professor: Paulo de Tarso F. Júnior  
[paulodt@gmail.com](mailto:paulodt@gmail.com)