



**USP - ICMC - SSC  
SSC 0501 - 1o. Semestre 2010**

**Disciplina de  
Introdução à Ciência da Computação  
ICC 1 - Teoria**

Prof. Fernando Santos Osório

Email: fosorio [at] { icmc. usp. br , gmail. com }

Página Pessoal: <http://www.icmc.usp.br/~fosorio/>

Web - WIKI ICMC: <http://wiki.icmc.usp.br/index.php/SSC-501>

PAE: Gustavo Pessin (Dout. CCMC) / <Http://pessin.googlepages.com/>

Email: pessin [at] { icmc.usp.br , gmail.com }

Monitor: Matheus Lin Alvarenga (EC) / <Http://matheuslin.wordpress.com/>

Email: matheus.lin [at] gmail.com

*Aula 04s*

1

Março 2010

**Linguagem de Programação “C”**

**Agenda:**

- **Recordar é viver...**
- **Programa com Desvios IF em “C”**
- **E a vida continua...**
- **Programas com Laços de Repetição: FOR**

**Informações Complementares e Atualizadas:**

**Consulte REGULARMENTE**

**O material disponível na COTEIA**

2

Março 2010

## Comando IF

### IF

```
if ( <expressão> )  
    <comando>;  
else  
    <comando>;
```

```
if ( salario > 100.00)  
    printf (“Salário maior que R$100,00\n”);
```

```
if ( salario == 0.00)  
    printf (“Este já foi demitido faz tempo...\n”);  
else  
    printf (“Este ainda está sendo pago...\n”);
```

**CUIDADO:**  $if(a == b)$  NÃO É O MESMO QUE  $if(a = b) !!$   
 $if(a > 18)$ ;  $\leftarrow printf(“maior!\n”); /* if “vazio” */$

### Expressão:

- Expressão lógica, relacional, aritmética

### Comando:

- Comando simples ou bloco de comandos

- Bloco de comandos: { ... }

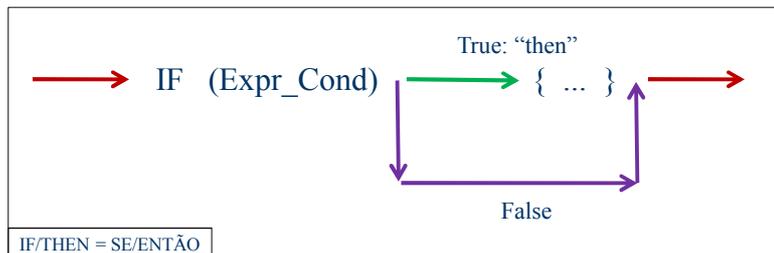
{ comando;comando; ... } ~ comando;

3

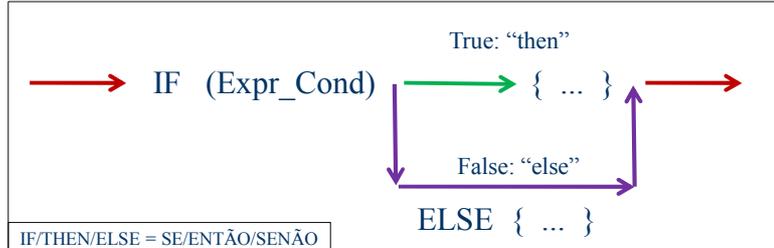
Março 2010

## Fluxo de um Programa em "C"

Fluxo de Execução de um Programa => **Desvios Condicionais (IF)**



IF/THEN = SE/ENTÃO



IF/THEN/ELSE = SE/ENTÃO/SENÃO

4

Março 2010

### Expressões Condicionais: Lógicas, Relacionais, Expressões Gerais

#### 1. Operadores Lógicos:

And (&&), Or (||) , Not (!)

#### 2. Operadores Relacionais

Comparação: >, >=, <, <=, !=, ==

#### 3. Operadores e Expressões em "C":

- Operadores de Atribuição,
- Operadores Aritméticos,
- Operadores Lógicos e Relacionais,
- Operadores Bitwise (bit a bit),
- Operadores de Assinalamento (var op= expr)
- Pré-Pos Incremento-Decremento ( var++, --var),
- Operadores de Endereço

### Fluxo de Execução de um Programa => **Desvios Condicionais (IF)**

#### Exercício:

Faça um programa adequado para o cálculo da média final (médias simples de 2 notas) de um aluno de uma disciplina indicando seu status final (aprovado/reprovado)

#### Cálculo da Média:

Ler as notas obtidas pelo aluno

Nota1: [0.0 .. 10.0]

Nota2: [0.0 .. 10.0]

Se alguma das notas estiver fora do intervalo especificado (entre 0.0 e 10.0) o programa deve indicar que houve um erro na nota e encerrar a execução.

Calcular a Média Final da Avaliação (MF)

$MF = (Nota1 + Nota2) / 2$

Se a nota está igual ou acima da média mínima (5.0), o aluno foi aprovado

Se a nota está abaixo da média mínima (5.0), o aluno foi reprovado



### Exercício:

Faça um programa adequado para o cálculo da nota final e da situação final de um aluno de uma disciplina avaliada como descrito previamente (aprovado/reprovado)

```
#include <stdio.h>
#include <stdlib.h>

double N1, N2;
double MFinal;

int main(int argc, char *argv[])
{
    printf("\n");
    printf("Nota 1: ");
    scanf ("%lf",&N1);
    if ( (N1 < 0.0) || (N1 > 10.0) )
    {
        printf(">> Erro: nota invalida!\n");
        system("PAUSE");
        return 1;
    }
    printf("Nota 2: ");
    scanf ("%lf",&N2);
    if ( (N2 < 0.0) || (N2 > 10.0) )
    {
        printf(">> Erro: nota invalida!\n");
        system("PAUSE");
        return 1;
    }

    /* Calcula a Media Final */
    MFinal = ( N1 + N2 ) / 2.0;

    /* Verifica se foi aprovado */
    if (MFinal >= 5.0)
        printf("\nAPROVADO=%.2lf!\n",MFinal);
    else
        printf("\nREPROVADO=%.2lf!\n",MFinal);

    printf("\n");
    system("PAUSE");
    return 0;
} /* fim programa */
```

### Agenda:

- Programas Seqüenciais x Programas com Repetição
- Comandos de Repetição (laços): FOR
  - > Fluxo de Execução com Laços FOR
  - > Comandos FOR: Contadores
  - > Comandos FOR: Acumuladores (Somadores)
  - > Comandos FOR: Aninhados
  - > Exercícios

**Informações Complementares e Atualizadas:**  
**Consulte REGULARMENTE**  
**O material disponível na COTEIA**

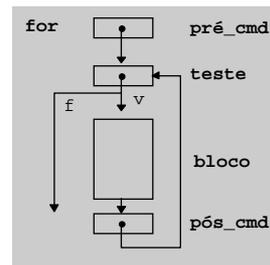


## Comando de Laço FOR

### Comando de Repetição: FOR

```
for (<pré_cmd>; <teste condição> ; <pós_cmd>)  
{  
  comandos;  
}
```

```
for (A = 1; A < 10 ; A++)  
{  
  printf ("%d \n", A);  
}  
printf ("FIM!\n");
```



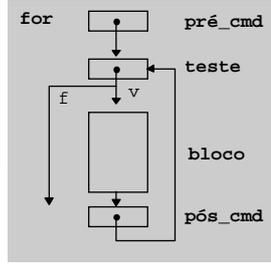
## Comando de Laço FOR

### Comando de Repetição: FOR

```
for (<pré_cmd>; <teste condição> ; <pós_cmd>)  
{  
  comandos;  
}
```

```
A = 1;  
if (A < 10)  
{  
  printf ("%d \n", A);  
  A = A + 1;  
  <Laço: Continue no IF>  
}  
printf ("FIM!\n");
```

```
for (A = 1; A < 10 ; A++)  
{  
  printf ("%d \n", A);  
}  
printf ("FIM!\n");
```



## Comando de Laço FOR

### FOR

```
for ( <expr_inicial>; <condição_de_parada>; <alteração_var_controle> )  
  <comando>;
```

```
for ( contador=0; contador < nro_vezes; contador++ )  
  printf("Contando... %d\n", contador);
```

```
for (contador=10; contador != 0; contador-- )  
{  
  printf("Contagem regressiva...\n");  
  printf("Falta: %d \n", contador);  
}
```

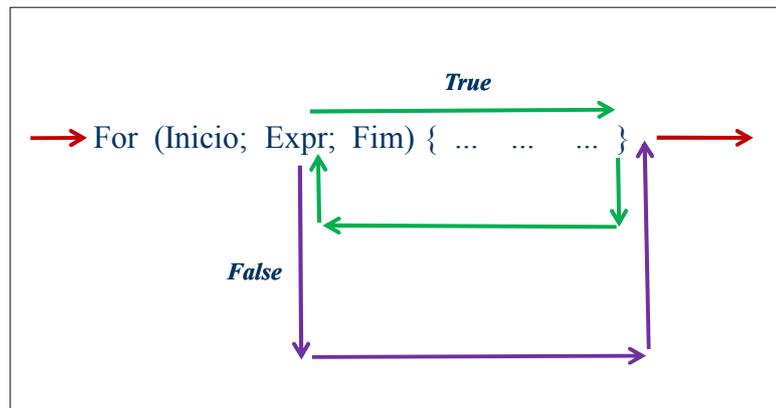
**CUIDADO:** for ( a=1; a <= 10; a++ ) ;  
for ( ; ; ) /\* Loop infinito \*/

15

Março 2010

## Fluxo de um Programa em "C": FOR

Fluxo de Execução de um Programa => Repetição FOR



```
for (<início>; <teste condição final>; <ajuste fim> )  
{ comandos; }
```

**FOR:** Para que a variável varie do valor de *início* até o valor de *final*

16

Março 2010

### Comando de Laço: FOR

- Estrutura de laço que permite realizar uma repetição até que uma condição seja satisfeita. Usado para fazer contadores.



```
for (A = 1; A < 10 ; A++)
{
    printf ("%d \n", A);
}
printf ("FIM!\n");
```

### Comando de Laço: FOR

- Comandos FOR são usados para contar...  
1, 2, 3, 4, ... N ( x = 1 ; x = x + 1 ; enquanto x <= N )  
10, 11, 12, ... M ( x = 10 ; x = x + 1 ; enquanto x <= M )  
10, 9, 8, 7, ... 1 ( x = 10 ; x = x - 1 ; enquanto x >= 1 )  
1, 3, 5, 7, 9 ... K ( x = 1 ; x = x + 2 ; enquanto x <= K )

- Comandos FOR são usados acumular...  
Somar as notas de 10 alunos (somatório)

$$N_1 + N_2 + N_3 + \dots + N_{10}$$

Multiplicar K valores (produtório)

$$V_1 * V_2 * V_3 * \dots * V_k$$

$$\sum_{i=1}^K N_i$$

### Comando de Laço: FOR

- Comandos FOR podem ser usados aninhados...

Para cada aluno da turma...

Para cada nota do aluno...

```
for (aluno = 1; aluno <= 30; aluno ++)  
{  
    Total=0;  
    for (aval = 1; aval <= 3; aval ++)  
    {  
        scanf("%lf",&Nota);  
        Total=Total + Nota;  
    }  
    printf ("Aluno %d - Media = %.2lf", aluno, Total/3.0 );  
}
```

19

Março 2010

### Comando de Laço: FOR

- Comandos FOR usados como laço sem fim...

**BREAK**

break; => Força o término da execução de um loop

```
for ( ; ; )  
{  
    printf ("Digite a Nota do Aluno: ");  
    scanf ("%lf",&Nota);  
    if ((Nota >= 0.0) && (Nota <= 10.0))  
        break;  
    printf("Nota invalida!\n");  
}
```

20

Março 2010

### Comando de Laço: FOR

- Comandos FOR usados como laço sem fim...

#### CONTINUE

continue; => Força o recomeço da execução de um loop

```
for ( ; ; )  
{  
    printf ("Digite a Nota do Aluno: ");  
    scanf ("%lf",&Nota);  
    if ((Nota < 0.0) || (Nota > 10.0))  
        continue;  
    printf ("Nota correta...\n");  
    break;  
}
```

### Comandos

```
if (<expressão> comando ;  
if (<expressão> { comandos }  
if (<expressão> comando ; else comando ;  
if (<expressão> { comandos } else { comandos }
```

```
for (<início>;<expr>;<fim>) comando ;  
for (<início>;<expr>;<fim>) { comando }
```

```
printf ("...", ... ) ;  
scanf ("...", ... ) ;
```

Onde e quando  
colocar o ; !?!

### EXERCÍCIOS: USANDO FOR

- Faça um programa que exiba na tela uma contagem regressiva de 10 até 0.
- Faça um programa onde o usuário determine os valores inicial e final de uma contagem (ler os valores inicial e final) e depois realizar uma contagem exibindo os valores na tela.  
*Nota:* O valor inicial pode ser menor que o final (contagem crescente) ou pode ser maior que o final (contagem decrescente).
- Faça um programa que calcule o fatorial de um número fornecido pelo usuário.  
*Nota:* Fatorial de  $N$  é  $N * (N-1) * (N-2) * \dots * 1 \Rightarrow N!$   
Fatorial de 0 é 1  $\Rightarrow 0!$  é igual a 1
- Faça um programa que leia 2 notas de cada aluno nas provas P1 e P2. As 2 notas devem ser válidas, ou seja, ter valores entre 0 e 10. Calcule a média ponderada destas 2 notas, onde a nota da P1 tem peso 1 e a nota da P2 tem peso 2, exibindo a média na tela. Repita este procedimento para uma turma de 10 alunos.

23

Março 2010

### COMPLEMENTO:

#### ➤ STRINGS

```
char Nome[30];  
char Letra;
```

```
scanf ("%s", Nome);  
strcpy (Nome, "Fulano");
```

	Nome
	[0] [1] [2] [3] [4] [5] [6] [7] [8] ... [29]
	'F' 'u' 'l' 'a' 'n' 'o' '\0' ? ? ?

```
Letra = 'Z';  
Nome[0] = Letra;  
Letra = 'o';  
Nome[1] = Letra;  
Nome[5] = 'a';
```

```
printf("%s",Nome);
```

➔ Vai escrever na tela: Zolana

*Nota:* Nome == &(Nome[0]) ➔ scanf("%s",&(Nome[0]));

24

Março 2010



**INFORMAÇÕES SOBRE A DISCIPLINA**

**USP - Universidade de São Paulo - São Carlos, SP**  
**ICMC - Instituto de Ciências Matemáticas e de Computação**  
**SSC - Departamento de Sistemas de Computação**

**Prof. Fernando Santos OSÓRIO**

**Web institucional: <http://www.icmc.usp.br/ssc/>**

**Página pessoal: <http://www.icmc.usp.br/~fosorio/>**

**E-mail: [fosorio \[at\] icmc. usp. br](mailto:fosorio@icmc.usp.br) ou [fosorio \[at\] gmail. com](mailto:fosorio@gmail.com)**

**PAE Gustavo Pessin – E-mail: [pessin \[at\] gmail .com](mailto:pessin@gmail.com)**

**Monitor Matheus Lin – E-mail: [matheus.lin \[at\] gmail.com](mailto:matheus.lin@gmail.com)**

**Disciplina de Introdução a Ciência da Computação**

**Web disciplina: Wiki ICMC - [Http://wiki.icmc.usp.br](http://wiki.icmc.usp.br)**

**> Programa, Material de Aulas, Critérios de Avaliação,**

**> Trabalhos Práticos, Datas das Provas, Notas**