

PROGRAMAÇÃO DE COMPUTADORES V - TCC- 00.323

## Modulo 3: Else, Case, For...

Aura - Erick  
[aconci@ic.uff.br](mailto:aconci@ic.uff.br), [erickr@id.uff.br](mailto:erickr@id.uff.br)

# Roteiro

---

- ▶ Mais Comandos Condicionais
- ▶ Estrutura de Repetição Contável



## Exemplo de uso do Comando Condicional

---

Média - M	Conceito -CONC
$M < 6$	D
$6 \leq M < 7.5$	C
$7.5 \leq M < 9$	B
$9 \leq M$	A



# Algoritmo

---

1. Leia **M**
2. Se **M** < 6  
Então **CONC** = "D"
3. Se **M** >= 6 AND **M** < 7.5  
Então **CONC** = "C"
4. Se **M** >= 7.5 AND **M** < 9  
Então **CONC** = "B"
5. Se **M** >= 9  
Então **CONC** = "A"



# Código em C... Com if

---

```
#include <stdio.h>
main ()
{
float M; /* a média */
char CONC; /* o conceito */
printf (" Emissao do conceito final dada a media\n");
printf (" Informe a media alcancada pelo aluno: ");
scanf ("%f ", &M);
if (M < 6.)
    CONC = 'D';
if (M >= 6. && M < 7.5)
    CONC = 'C';
if (M >= 7.5 && M < 9.)
    CONC = 'B';
if (M >= 9.)
    CONC = 'A';

printf (" o conceito final do aluno: %c", CONC);
}
```

---



## Algoritmo anterior x Algoritmo Alternativo

---

1. Leia M
2. Se  $M < 6$   
Então CONC = "D"
3. Se  $M \geq 6$  AND  $M < 7.5$   
Então CONC = "C"
4. Se  $M \geq 7.5$  AND  $M < 9$   
Então CONC = "B"
5. Se  $M \geq 9$   
Então CONC = "A"

1. Leia M
2. Se  $M < 6$   
Então CONC = "D"  
Senão Se  $M < 7.5$   
Então CONC = "C"  
Senão Se  $M < 9$   
Então CONC = "B"  
Senão CONC = "A"



## Algoritmo anterior x Algoritmo Alternativo

1. Leia M
2. Se  $M < 6$   
Então CONC = "D"
3. Se  $M \geq 6$  AND  $M < 7.5$   
Então CONC = "C"
4. Se  $M \geq 7.5$  AND  $M < 9$   
Então CONC = "B"
5. Se  $M \geq 9$   
Então CONC = "A"

1. Leia M
2. Se  $M < 6$   
Então CONC = "D"  
Senão Se  $M < 7.5$   
Então CONC = "C"  
Senão Se  $M < 9$   
Então CONC = "B"  
Senão CONC = "A"

Mais  
comparações

Mais eficiente

# indentação

---

▶ é o recuo de um texto em relação a sua margem.

▶ [Dúvidas de Português](#) >

**Identação** ou **indentação** ?

A forma correta de escrita da palavra é [indentação](#).

A palavra **identação** está **errada**.

A palavra **indentação** é um neologismo que tem sua origem na palavra inglesa **indentation**, tendo um sentido de recuo.

É muito usada na área informática, em linguagem de programação.

---



# Em C... Com If ... else

---

```
#include <stdio.h>
void main ()
{
float M; /* a média */
char CONC; /* o conceito */
printf (" Emissao do conceito final dada a media \n");
printf (" Informe a media alcancada pelo aluno: ");
scanf ("%f", &M);
if (M < 6.)
    CONC = 'D';
else if (M < 7.5)
    CONC = 'C';
    else if (M < 9.)
        CONC = 'B';
        else CONC = 'A';

printf (" o conceito final do aluno: %c", CONC);
}
```

---



## Exercício

---

- ▶ Faça um programa em C que lê 2 números inteiros (N1, N2) e retorna o maior deles.
- ▶ Use a estrutura IF-ELSE.



# Estrutura de seleção múltipla

---

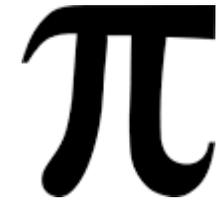
- ▶ Em alguns casos, o comportamento do programa varia de acordo com um critério de seleção que tem várias opções
- ▶ Exemplo: um programa que tem comportamentos diferentes, dependendo da operação escolhida pelo usuário
  - ▶ Ler OPERACAO e um valor X
  - ▶ Se OPERACAO = 1 → calcular a **área da circunferência** de raio x
  - ▶ Se OPERACAO = 2 → calcular o **perímetro da circunferência** de raio x



# Algoritmo

---

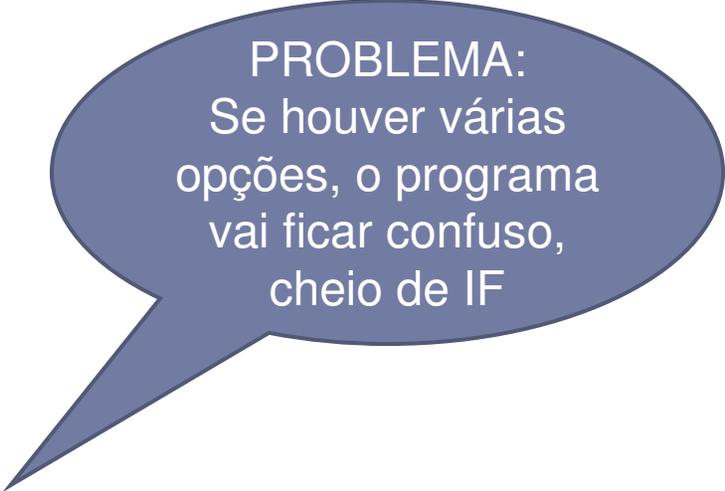
1. Leia OP, X
2. Se  $OP = 1$   
Então calcula área  
Imprime área
3. Se  $OP = 2$   
Então calcula perímetro  
Imprime perímetro
4. Se  $OP \neq 1$  e  $OP \neq 2$   
Então imprime “erro”



# Algoritmo

---

1. Leia OP, X
2. Se  $OP = 1$   
Então calcula área  
Imprime área
3. Se  $OP = 2$   
Então calcula perímetro  
Imprime perímetro
4. Se  $OP \neq 1$  e  $OP \neq 2$   
Então imprime “erro”  
(pois não é uma opção válida)



PROBLEMA:  
Se houver várias opções, o programa vai ficar confuso, cheio de IF



# Solução: Estrutura de **Seleção Múltipla**

---

SELECIONE <seletor> de  
<valor 1> : <comando>  
<valor 2> : <comando>  
...  
<valor n> : <comando>  
[senão <comando>]  
FIM SELEÇÃO



## Em C...

---

```
switch (expr)
{
    case const1: comando1;
    break;
    case const2: comando2;
    break;
    ...
    default: comando3;
}
```

expr

- Uma expressão que retorna um valor

const

- Valor numérico
- Caractere

comando

- Simples
  - Composto (entre chaves)
- 
- 

# Funcionamento

---

1. A expressão é avaliada;
2. O resultado da expressão é comparado com os valores das constantes que aparecem nos comandos case;
3. Quando o resultado da expressão for igual a uma das constantes, a execução se inicia a partir do comando associado com esta constante.

A execução continua com a execução de todos os comandos até o fim do comando **switch**, ou até que um comando **break** seja encontrado.

---



## Funcionamento (cont.)

---

4. Caso não ocorra nenhuma coincidência o comando **default** é executado. O comando **default** é opcional e se ele não aparecer nenhum comando será executado.
  - ▶ O comando **break** é um dos comandos de desvio da linguagem C.
  - ▶ O **break** é usado dentro do comando **switch** para interromper a execução e **pular para o comando seguinte** ao comando **switch**.

# Exemplo em C

---

```
#include <stdio.h>
#define PI=3.14159265359
main ()
{
    int OP, X;
    float RESULTADO;
    scanf("%d", &X);
    scanf("%d", &OP);
    switch (OP) {
        // calcula perimetro
        case 1: RESULTADO = 2 * PI * X;
                printf("O perimetro eh %f", RESULTADO);
                break;
        // calcula area
        case 2: RESULTADO = PI * X * X;
                printf("A area eh %f", RESULTADO);
                break;
        default: printf("Operacao errada");
    }
}
```

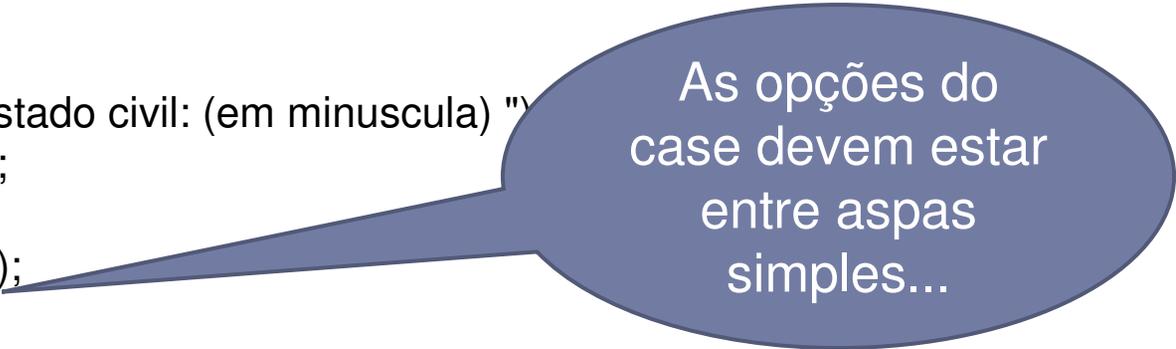
---



# Estado Civil – usando switch com **char**

---

```
#include <stdio.h>
main ()
{
    char EstadoCivil;
    printf("Digite a inicial do estado civil: (em minuscula) ");
    scanf("%c", &EstadoCivil);
    switch (EstadoCivil) {
        case 'c': printf("Casado");
                break;
        case 'd': printf("Divorciado");
                break;
        case 's': printf("Solteiro");
                break;
        case 'v': printf("Viuvo");
                break;
        default: printf("Estado civil desconhecido");
    }
    system("PAUSE");
}
```



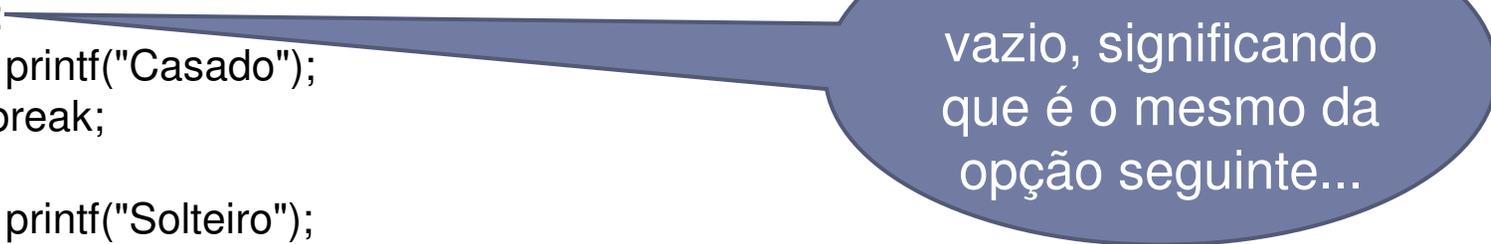
As opções do case devem estar entre aspas simples...



# Estado Civil – maiúsculas e minúsculas

---

```
#include <stdio.h>
main ()
{
    char EstadoCivil;
    printf("Digite a inicial do estado civil: ");
    scanf("%c", &EstadoCivil);
    switch (EstadoCivil) {
        case 'C':
        case 'c': printf("Casado");
                break;
        case 'S':
        case 's': printf("Solteiro");
                break;
        case 'V':
        case 'v': printf("Viuvo");
                break;
        default: printf("Estado civil desconhecido");
    }
    system("PAUSE");
}
```



Comando fica vazio, significando que é o mesmo da opção seguinte...



# Exercício

- ▶ Fazer o fluxograma de um programa que pergunta as coordenadas de 2 pontos do plano e imprima **a distância entre eles ao quadrado**. Cada ponto deve ser localizado por uma coordenada **horizontal** e outra **vertical**. Depois calcule a distância entre os dois pontos considerando a expressão (denominada distância Euclidiana ao quadrado) :

- ▶  $H \times H + V \times V$

Onde H e V é **a diferença entre as coordenadas horizontais e verticais** dos dois pontos. Mas caso **apenas um dos dois pontos** esteja **sobre o eixo vertical** de coordenadas, **essa coordenada deve ser somada de 1**, antes de ser feito o cálculo acima.

Caso as coordenadas verticais (ou horizontais) de ambos os pontos sejam zero a distancia entre os dois pontos é dada por:

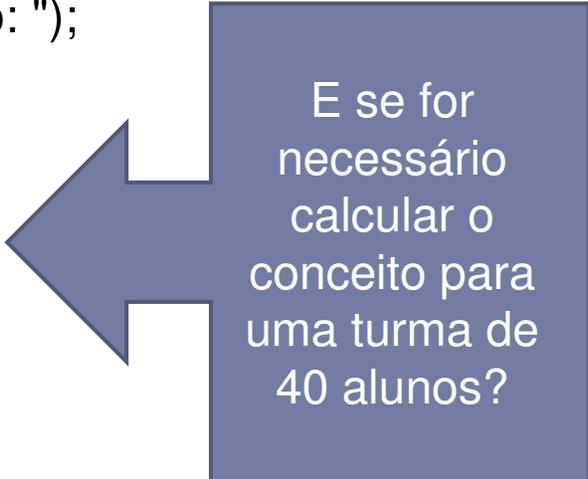
- ▶  $H \times H$ , onde H é **a diferença entre as coordenadas horizontais**. Ou  $V \times V$  onde V é **a diferença entre as coordenadas verticais**.
- ▶ Imprima o resultado desta distância (ao quadrado) entre os pontos que chamaremos de **distância D**.



# Repetições

---

```
#include <stdio.h>
void main ()
{
float M; /* a média */
char CONC; /* o conceito */
printf ("#Emissao do conceito final dada a media\n");
printf ("#Informe a media alcancada pelo aluno: ");
scanf ("%f", &M);
if (M < 6.)
    CONC = 'D';
else if (M < 7.5)
    CONC = 'C';
    else if (M < 9.)
        CONC = 'B';
        else CONC = 'A';
printf ("Conceito %c", CONC);
}
```



E se for necessário calcular o conceito para uma turma de 40 alunos?



## Solução: Repetição contável

---

Para VAR\_CONTROLE de 1 até 100 faça  
<comando>

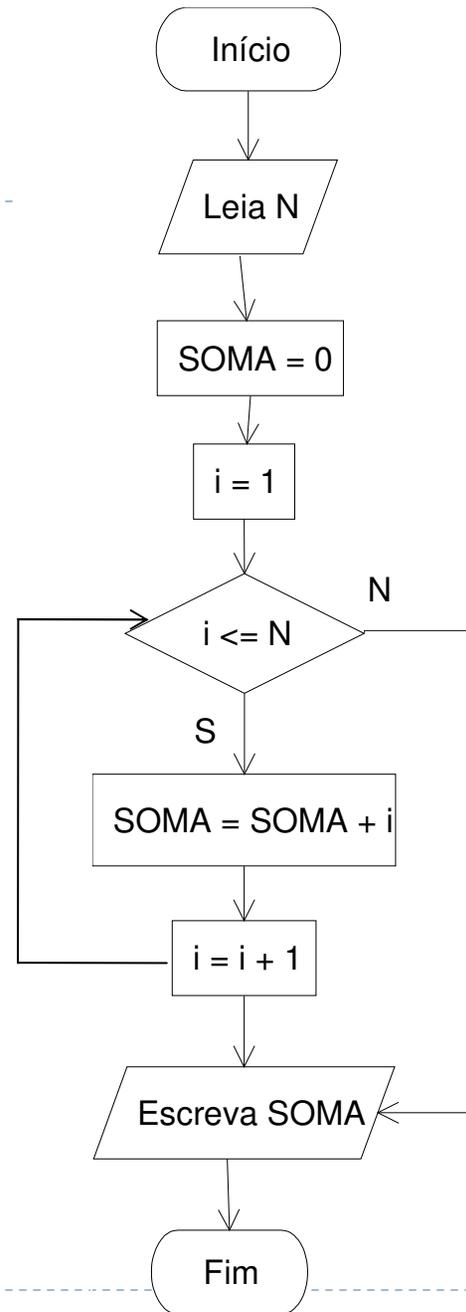
**Como você escreveria o Google => O número 10 elevado a potência 100 ( $10^{100}$ ), que pode ser escrito como o numeral 1 seguido de 100 zeros.**



## Exemplo:

---

- ▶ Calcular a soma dos  $N$  primeiros naturais



## Em C...

---

for (expressão1; expressão2; expressão3) comando;

- ▶ As três expressões geralmente têm os seguintes significados:
  - ▶ A expressão1 é utilizada para **inicializar** a variável de controle do laço;
  - ▶ A expressão2 é um **teste** que controla o fim do laço;
  - ▶ A expressão3 normalmente faz um **incremento ou decremento** da variável de controle.



# Funcionamento

---

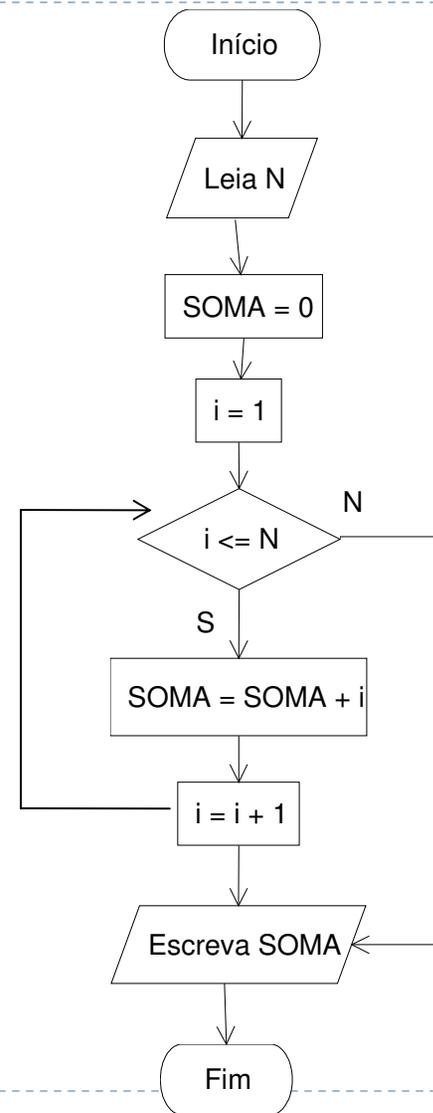
A execução do comando **for** segue os seguintes passos:

1. A expressão1 é avaliada;
2. A expressão2 é avaliada para determinar se o comando deve ser executado;
3. Se o resultado da expressão2 for VERDADEIRA o comando é executado caso contrário o laço é terminado;
4. A expressão3 é avaliada;
5. Voltar para o passo onde 2.



# Soma dos N primeiros números naturais

```
#include <stdio.h>
main ()
{
    int N, i, SOMA;
    SOMA = 0;
    printf("Digite um numero natural: ");
    scanf("%d", &N);
    for (i=0; i<=N; i++)
    {
        SOMA = SOMA + i;
    }
    printf("A soma eh %d", SOMA);
    system("PAUSE");
}
```



# Ler 5 números inteiros e contar quantos são negativos

---

```
#include <stdio.h>
#include <stdlib.h>
main()
{
    int numero, cont, neg=0;
    for (cont=0; cont<5; cont++)
    {
        printf ("\nDigite um numero inteiro: ");
        scanf ("%d", &numero);
        if (numero<0)
            neg++;
    }
    printf ("\nO numero de valores negativos eh %d", neg);
    system("pause");
}
```



# for dentro de for

---

```
#include <stdio.h>
main ()
{
    int i,j;
    for(i=0;i<=10;i++) {
        for(j=0;j<=2;j++) {
            printf("%d %d \n", i,j);
        }
    }
    system("PAUSE");
}
```

Saída:

```
0 0
0 1
0 2
1 0
1 1
1 2
2 0
2 1
2 2
...
10 0
10 1
10 2
```



# Exercícios

---

- ▶ Fazer o algoritmo e código que imprima o número google.
- ▶ Fazer o código que calcule a distância euclidiana,  $E$ , e a  $D$ , definida anteriormente, entre 2 pontos do plano fornecidos pelo usuário, por meio de suas coordenadas horizontais e verticais :  $(H1, V1)$   
 $(H2, V2)$ .

