

Cláudio C. Rodrigues
Faculdade da Computação - UFU

◦ **LINGUAGEM C - 02: ESTRUTURA DE SELEÇÃO CONDICIONAL**

Objetivo

Apresentar os conceitos básicos e as estruturas de linguagem relacionadas à tomada de decisão ofertadas pela linguagem C, tais como:

- Valores lógicos
- Operadores relacionais
- Operadores lógicos
- **Comandos de Seleção Condicional**

Estrutura Condicional

1. Expressões Lógicas
 - Operador Relacional
 - Operador Lógico
2. Estrutura Condicional Simples
3. Estrutura Condicional Composta
4. Operador Condicional
5. Condicionais Aninhados e Encadeados
6. Estrutura Case

1. Expressões Lógicas

- Em **C**, não existe um tipo específico para a representação de valores lógicos.
- Entretanto, qualquer valor pode ser interpretado como um valor lógico:
 - **zero** representa *falso* e qualquer **outro valor** representa *verdadeiro*

Operadores Relacionais

- Para gerar um *valor lógico*, usamos os *operadores relacionais*.

Operador Relacional	Resultado
$X == Y$	Verdade se X for igual a Y
$X != Y$	Verdade se X for diferente de Y
$X < Y$	Verdade se X for menor que Y
$X > Y$	Verdade se X for maior que Y
$X <= Y$	Verdade se X for menor ou igual a Y
$X >= Y$	Verdade se X for maior ou igual a Y

Operadores Lógicos

- Além dos *operadores relacionais*, **C** oferece também *operadores lógicos*. Com eles, podemos criar expressões lógicas compostas

Operador Lógico	Resultado
$!x$	Verdade se e somente se x for falso
$x \ \&\& \ y$	Verdade se e somente se x e y forem verdadeiros
$x \ \ y$	Verdade se e somente se x ou y for verdadeiro

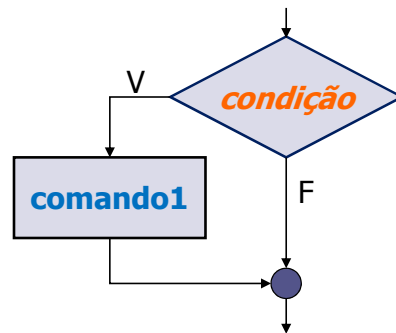
- **Qual a saída produzida pela instrução abaixo?**

```
printf("%d %d %d %d", !3, !0, 3+'a'>'b'+2 && !'b', 1 || !2 && 3);
```

Estrutura Condicional Simples

- A estrutura condicional de decisão simples serve para escolher um comando para execução, caso a condição for verdadeira.
- Sintaxe da **estrutura condicional simples**:

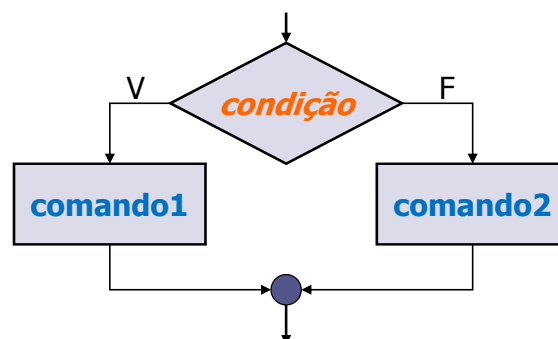
```
if( condição ) comando1;
```



Estrutura Condicional Composta

- A estrutura condicional ou de decisão composta serve para escolher um entre dois blocos de comandos alternativos.
- A **estrutura condicional composta** é codificada como:

```
if( condição ) comando1; else comando2;
```



Programa exemplo 1:

```
#include <stdio.h>
int main()
{
    float a, b, m;
    printf("\n Digite duas notas : ");
    scanf("%f %f", &a, &b);
    m = (a+b)/2;
    if( m >= 7.0 ) printf("\n Aprovado");
    else printf("\n Reprovado");
    return 0;
}
```

Desafios de Programação:

1. **Dado um ano, informe se ele é ou não bissexto.**
[Dica: um ano é bissexto se é divisível por 4 mas não por 100]
2. **Em uma empresa paga-se R\$ 19,50 a hora e recolhe-se para o imposto de renda 10% dos salários acima de R\$ 1500,00. Dado o número de horas trabalhadas por um funcionário, informar o valor do seu salário líquido.**

Operador Condicional

- **C** oferece também um operador que proporciona uma forma mais compacta de se representar decisões simples. O **operador condicional**, cuja sintaxe é

condição ? expressão1 : expressão2

- Exemplo:

`abs = n > 0 ? n : -n;`

Uma vantagem no uso do operador condicional é que, sendo um operador, podemos utilizá-lo em qualquer contexto em que uma expressão é permitida.

Condicionais Aninhados e Encadeados

- estrutura condicional serve para selecionar e executar um entre dois comandos alternativos.
- Nesse caso, dizemos que o primeiro condicional é o *principal* e o outro está *aninhado* ou *encadeado*.

- Algumas vezes, um destes comandos alternativos (ou ambos) sejam também condicionais.

- Sintaxe:

```
if( condição ) /* principal */  
  if ...      /* aninhado */  
else  
  if ...      /* encadeado */
```

Programa exemplo 2:

- **Problema:**

“Dados três números verificar se eles podem representar as medidas dos lados de um triângulo e, se puderem, classificar o triângulo em equilátero, isósceles ou escaleno”.

- Para que três números representem os lados de um triângulo é necessário que cada um deles seja menor que a soma dos outros dois.
- Um triângulo é equilátero se tem os três lados iguais, isósceles se tem apenas dois lados iguais e escaleno se tem todos os lados distintos

```
#include <stdio.h>

int main() {
    float a, b, c;
    printf("\nDigite três números: ");
    scanf("%f %f %f", &a, &b, &c);
    if( a<b+c && b<a+c && c<a+b ) {
        printf("\nÉ um triângulo: ");
        if( a==b && b==c ) printf("equilátero");
        else if( a==b || a==c || b==c ) printf("isósceles");
        else printf("escaleno");
    }
    else printf("\nNão é um triângulo");
    return 0;
}
```

Desafio de Programação

P1: Numa faculdade, os alunos com média maior ou igual a 7,0 são aprovados, aqueles com média inferior a 3,0 são reprovados e os demais ficam de recuperação. Dadas as duas notas de um aluno, informe sua situação. Emitir as mensagens aprovado, reprovado e recuperação, respectivamente

P2: Dados os coeficientes ($a \neq 0$, b e c) de uma equação do 2º grau, calcule e informe suas raízes reais, usando a fórmula de Báskara:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Estrutura Switch-Case

- A **estrutura de decisão múltipla** é bastante adequada quando precisamos escolher uma entre várias alternativas previamente definidas, por exemplo, num menu.
- A **estrutura de decisão múltipla** tem a seguinte sintaxe:

```
switch( expressão )
{
    case constante1 : comando1; break;
    case constante2 : comando2; break;
    ...
    case constanten : comandon; break;
    default          : comando;
}
```


Programa exemplo 3:

```
#include <stdio.h>
int main() {
    int n;
    printf("\n Digite um número: ");
    scanf("%d", &n);
    switch( n ) {
        case 1: putchar('A'); break;
        case 3: putchar('B');
        case 4: putchar('C'); break;
        default: printf('*');
        case 5: putchar('D');
    }
    putchar('.');
    return 0;
}
```

Bibliografia

- Pereira, S.L., Linguagem C – Distribuição gratuita
- Schildt, H., C Completo e Total, Editora Makron Books do Brasil Editora Ltda, 1996.
- Evaristo, J., Aprendendo a programar programando em linguagem C, Book Express, 2001.
- Mizrahi, V.V., Treinamento em Linguagem C, Curso Completo, Módulos 1 e 2, Makron Books do Brasil Editora Ltda, 1990.
- Kernighan, B.W & Ritchie, D. M., C a Linguagem de Programação, Editora Campus, 1986.