




INTRODUÇÃO À PROGRAMAÇÃO

Aula 8 - Estruturas Condicionais (Parte 2)

Prof. Carlos Alexandre Siqueira da Silva



Campus de Alegre



Revisão da Aula Anterior

- Estrutura Condicionais de programação utilizando a linguagem C
 - Expressão lógica.
 - Operadores Relacionais.
 - Comando **if**.
 - Comando **if-else**.

Operadores Lógicos

Esses são os operadores lógicos utilizados para unir duas expressões lógicas em uma condição composta:

Operador	Descrição
&&	Operador lógico "E"
	Operador lógico "OU"
!	Operador lógico "NÃO"

Tabela Verdade

A Tabela Verdade é uma ferramenta fundamental para entender o funcionamento dos operadores lógicos. Ela mostra todos os possíveis valores de entrada (**Verdadeiro** ou **Falso**) e o resultado final da expressão.

E1	E2	E1 && E2	E1 E2	!E1	!E2
V	V	V	V	F	F
V	F	F	V	F	V
F	V	F	V	V	F
F	F	F	F	V	V

Observação: Na tabela acima, "E1" e "E2" representam expressões lógicas que serão avaliadas pela estrutura condicional.

Condições Compostas

- Estruturas condicionais que avaliam duas ou mais condições em um mesmo bloco de comandos, utilizando operadores lógicos.

Sintaxe:

```
if((expressão lógica 1) && (expressão lógica 2)) { → Operador "E"  
    Bloco de comandos 1  
}  
if((expressão lógica 1) || (expressão lógica 2)) { → Operador "OU"  
    Bloco de comandos 2  
}  
if( !(expressão lógica)) { → Operador "NÃO"  
    Bloco de comandos 3  
}
```

Condições Compostas - Exemplo

Exemplo 1 – Verificar pela idade se a pessoa é um adulto:

```
1  if ((idade >= 18) && (idade <= 65)) {  
2      printf("Adulto\n");  
3  }
```

Condições Aninhadas

- Estruturas condicionais organizadas em **níveis** (uma dentro da outra), em que a expressão interna só é avaliada se o programa passar pela expressão externa.

Sintaxe:

```
if(expressão lógica 1) {  
    if(expressão lógica 2){  
        Bloco de comandos 1  
    }  
    if(expressão lógica 3){  
        Bloco de comandos 2  
    }  
}
```

Condições Aninhadas - Exemplo

Exemplo 2 – Verificar pela nota a situação do estudante:

```
1  if (nota >= 60) {  
2      printf("Aprovado\n");  
3  }  
4  else {  
5      if (nota >= 50) {  
6          printf("Recuperacao\n");  
7      }  
8      else {  
9          printf("Reprovado\n");  
10     }  
11 }
```


Estrutura de múltiplas escolhas (switch-case)

- Estruturas condicionais que permitem selecionar uma dentre várias opções possíveis.

Sintaxe:

```
switch (Variavel) {  
    case 1: → Primeira opção  
        Bloco de comandos 1  
        break;  
    case 2: → Segunda opção  
        Bloco de comandos 2  
        break;  
    default: → Opção padrão, caso não seja nenhuma das opções acima  
        Bloco de comandos padrao  
}
```

Estrutura de múltiplas escolhas - Exemplo

Exemplo 3 – Escolher uma operação bancária:

```
1  int opcao;  
2  printf("1 – Saque\n");  
3  printf("2 – Deposito\n");  
4  scanf("%d", &opcao);  
5  switch (opcao) {  
6      case 1:  
7          printf("Saque em conta corrente\n");  
8          break;  
9      case 2:  
10         printf("Deposito em conta corrente\n");  
11         break;  
12     default:  
13         printf("Opcao invalida\n");  
14 }
```

Desafio Anterior - Área do Quadrado ou Retângulo

```
1  #include <stdio.h>
2
3  int main()
4  {
5      float Base, Altura, Area;
6      printf("Digite a base e a altura:\n");
7      scanf("%f %f", &Base, &Altura);
8      Area = Base * Altura;
9      printf("%.1f\n", Area);
10     if(Base == Altura){
11         printf("Quadrado");
12     }
13     else{
14         printf("Retangulo");
15     }
16     return 0;
17 }
```

O que vem por aí...

Na próxima aula, vamos conhecer as **estruturas de repetição** da programação, entendendo a sua importância para os nossos programas.