

INTRODUÇÃO À PROGRAMAÇÃO

Prof.^a Priscilla Abreu

priscilla.braz@rj.senac.br



Introdução à Programação



Roteiro de Aula

- Objetivo da aula
- Revisão de conteúdo
- Estruturas de controle
 - Estrutura condicional
- Exercício

Introdução à Programação



Objetivo da aula

Compreender o funcionamento da estrutura condicional.

MAIS OPERADORES...

Introdução à Programação



OPERADORES RELACIONAIS

Comparam dois valores de um mesmo tipo básico. O resultado obtido é sempre um valor lógico.

Operador	Descrição
==	Igual
!=	Diferente
>	Maior que
>=	Maior ou igual que
<	Menor que
<=	Menor ou igual que

Introdução à Programação



OPERADORES LÓGICOS

São utilizados no processo de tomada de decisão e em controles de repetição.

O resultado da operação é sempre um valor lógico (VERDADEIRO (1) ou FALSO (0)).

São eles:

- | | |
|-----------------------|-------------|
| && (e) | - conjunção |
| (ou) | - disjunção |
| ! (não) | - negação |

OPERADORES LÓGICOS

Operador &&: a conjunção de dois ou mais operandos será verdadeira apenas quando todos os operandos possuírem o valor verdadeiro.

A	B	A && B
V	V	V
V	F	F
F	V	F
F	F	F

Introdução à Programação



OPERADORES LÓGICOS

Operador `||`: a disjunção de dois ou mais operandos será verdadeira sempre que pelo menos um dos operandos possuir o valor verdadeiro.

A	B	A B
V	V	V
V	F	V
F	V	V
F	F	F

Introdução à Programação



OPERADORES LÓGICOS

Operador ! : é um operador unário que inverte o valor do operando.

A	!(A)
V	F
F	V

OPERADORES LÓGICOS

Considere as seguintes variáveis:

A = 4, B = -2, C=1, TESTE = 1 e SEXO = 'M'

Qual o resultado das expressões?

- `A + B == 0 && C != 1`
- `(TESTE==1) || A * C > B`
- `!(TESTE)==1 && SEXO == 'F'`

Introdução à Programação



PRIORIDADE DAS OPERAÇÕES

Prioridade	Operador
1ª	Parênteses
2ª	Aritmético
3ª	Relacional
4ª	Lógicos

ESTRUTURA CONDICIONAL

Introdução à Programação



CONSIDERE ESSE EXEMPLO...

```
#include <stdio.h>
int main(){
    float nota1, nota2, media;
    printf("Informe a primeira nota:");
    scanf("%f", &nota1);
    printf("Informe a segunda nota:");
    scanf("%f", &nota2);
    media=(nota1+nota2)/2;
    printf("Média: %.1f", media);
}
```

Agora imagine que, além do cálculo da média, fosse necessário informar se o aluno está aprovado ou reprovado.

Como fazer esse programa?

Introdução à Programação



ESTRUTURAS DE CONTROLE

Diversas situações precisam de desvios ou análise para serem solucionadas, podendo apresentar resultados e caminhos diferentes dependendo de cada caso e valores considerados.

Introdução à Programação



ESTRUTURAS DE CONTROLE

Estruturas de controle controlam o fluxo de execução de um algoritmo.

Temos três categorias principais:

- Sequencial
- Seleção
- Iteração

Introdução à Programação



ESTRUTURAS DE CONTROLE

Uma das principais estruturas de seleção é a estrutura condicional **IF**.

A estrutura condicional divide-se em:

- Condicional Simples
- Condicional Composta

Introdução à Programação



ESTRUTURAS DE CONTROLE

Estrutura Condicional Simples

A instrução **if (condição)** tem por finalidade tomar uma decisão. Sendo a condição verdadeira, serão executadas todas as instruções que estejam associadas à instrução **if (condição)**.

Introdução à Programação



ESTRUTURA DE SELEÇÃO

Estrutura de Seleção: condicional simples

Sintaxe:

```
if (condição) {  
    comandos  
}
```

Introdução à Programação



ESTRUTURA DE SELEÇÃO

Estrutura de Seleção: condicional simples

Exemplo:

```
if (numero > 0){  
    printf("Número positivo!!!")  
}
```

ESTRUTURAS DE CONTROLE

Estrutura Condicional Composta (if / else)

- A seleção de uma ação é feita a partir da especificação de pelo menos duas alternativas e é dirigida por uma condição.
- Se a condição for verdadeira, um conjunto de ações será tomado; caso contrário, outro conjunto de ações será tomado.
- Com isso, o fluxo de condição é direcionado por uma tomada de decisão e alguns passos podem não ser processados, devido a um desvio.

Introdução à Programação



ESTRUTURA DE SELEÇÃO

Estrutura de Seleção: condicional composta

Sintaxe:

```
if(condição){  
    comandos  
}  
else{  
    comandos  
}
```

Introdução à Programação



ESTRUTURA DE SELEÇÃO

Estrutura de Seleção: condicional composta

Exemplo:

```
if(numero >= 0){  
    printf("Número positivo!")  
}  
else{  
    printf("Negativo!")  
}
```

Introdução à Programação



ESTRUTURA DE SELEÇÃO

Estrutura de Seleção: condicional composta

Exemplo:

```
if (numero > 0){  
    printf("Número positivo!")  
}  
else{  
    if (numero == 0)  
        printf("ZERO!");  
    else  
        printf("Negativo!")  
}
```


VOLTANDO À SITUAÇÃO INICIAL...

Introdução à Programação



ESTRUTURA SEQUENCIAL – EXEMPLO

```
#include <stdio.h>
int main(){
    float nota1, nota2, media;
    printf("Informe a primeira nota:");
    scanf("%f", &nota1);
    printf("Informe a segunda nota:");
    scanf("%f", &nota2);
    media=(nota1+nota2)/2;
    printf("Média: %.1f", media);
}
```

Introdução à Programação



ESTRUTURA SEQUENCIAL – EXEMPLO

```
#include <stdio.h>
int main(){
    float nota1, nota2, media;
    printf("Informe a primeira nota:");
    scanf("%f", &nota1);
    printf("Informe a segunda nota:");
    scanf("%f", &nota2);
    media=(nota1+nota2)/2;
    printf("Média: %.1f", media);
    if (media >= 7.0)
        printf("Aprovado!");
    else
        printf("Reprovado!");
}
```

Introdução à Programação



ESTRUTURA CONDICIONAL – EXERCÍCIOS

1. Escreva um programa em C que leia um número inteiro e informe se o número é par ou ímpar.

DÚVIDAS???