

Projeto de Lógica para programação

Grupo Pet-Computação
Matheus Freitag e Renata Junges

Comandos de controle

- Os comandos de controle permitem alterar a direção tomada por um programa (**desvio**), ou fazer com que partes específicas de um algoritmo seja executada mais de uma vez (**loop**).

- **Desvio Condicional:**

Muitas vezes será necessário desviar a execução do programa segundo uma condição. (Exemplo: ir a universidade de carro **ou** de ônibus?). Para se testar condições é necessário utilizar **operadores lógicos e relacionais**.

❏ Operadores Relacionais:

Permitem realizar a comparação de conteúdo das variáveis.

A **desigualdade** é dada por **!=** ;

A **igualdade** é dada por **==** ;

Maior que, pelo símbolo **>** ;

Maior ou igual, pelo símbolo **>=** ;

Menor que, pelo símbolo **<** ;

Menor ou igual, pelo símbolo **<=** ;

Desvio condicional (If-else)

Sintaxe:

```
if ( variavel1 > variavel2){  
    //Bloco de Instruções;  
}  
else{  
    //Bloco de Instruções;  
}
```

A instrução **if-else** é usada para fazer um desvio condicional. Ou seja, se a *comparação* que está dentro dos parênteses for *verdadeira*, será executada o bloco de instruções referentes ao **if**, senão, é realizada a execução do bloco de instruções referente ao **else**.

Exemplo de exercício com desvio condicional:

1. Dado dois números, mostre qual é o maior.

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

```
int main(){
```

```
    int x, y;
```

```
    printf("Informe o primeiro número:");
```

```
    scanf ("%d", &x);
```

```
    printf("Informe o segundo número:");
```

```
    scanf ("%d", &y);
```

```
        if (x>y){
            printf("O maior
número é: %d", x);
        }
        else{
            if (x<y){
                printf("O maior
número é: %d", y);
            }
            else printf ("São
iguais");
        }
        return 0;
    }
```

❏ Os operadores lógicos:

Os operadores "e", "ou" e "não" permitem realizar a combinação lógica de variáveis do tipo booleana (lógico).

Eles servem pra fazer a *comparação entre duas comparações* ou a *negação de uma variável* se o valor lógico for **true** (verdadeiro) um bloco de instruções será executado, se o valor lógico for **false** (falso), o outro bloco de instruções será executado.

Operador “e” = **&&**;

Operador “ou” = **||**;

Operador “not” = **!** ;

Tabela verdade

$(a > b) \ \&\& \ (c == d)$

V	V	---->	V
V	F	---->	F
F	V	---->	F
F	F	---->	F

$(a > b) \ || \ (c == d)$

V	V	---->	V
V	F	---->	V
F	V	---->	V
F	F	---->	F

$!(a > b)$

V	->	F
F	->	V

Exercício:

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

```
int main(){
```

```
    int a,b;
```

```
    printf ("Informe dois números: ");
```

```
    scanf("%d %d", &a, &b);
```

```
    if (a%2==0 && b%2==0){
```

```
        printf("Ambos são pares");
```

```
    }
```

```
    else if (( a%2==0 && !(b%2)==0) || ( !(a%2==0) && b%2==0)){
```

```
        printf(" Apenas um é par");
```

```
    }
```

```
    else printf ("Ambos são ímpares");
```

```
    return 0;
```

```
}
```

Escreva um algoritmo para ler dois números inteiros e escrever se ambos são pares, se ambos são ímpares, ou se um é par e o outro é ímpar.

Exercícios de fixação:

1. Calcular a média aritmética de três notas de um aluno e exibir a mensagem de APROVADO caso a média seja maior ou igual a 7 ou a mensagem REPROVADO caso a média seja menor que 7.
2. Escreva um programa para ler 3 valores, depois coloque em ordem crescente usando a instrução if-else. Considere que o usuário não informará valores iguais.

Grupo Pet-Computação

Matheus Freitag e Renata Junges