



Estrutura Condicional ou Estrutura de Desvio



Algoritmos e Programação A





Estrutura Condicional ou de Desvio

- A estrutura de desvio possibilita desviar a execução de instruções conforme o valor de um dado.



Estrutura Condicional ou de Desvio

```
1  #include <stdio.h>
2
3  int main(){
4      float a, b, c;
5
6      printf("**** Divisão de Dois Números ****\n");
7      printf("Digite o primeiro número: ");
8      scanf("%f", &a);
9      printf("Digite o segundo número: ");
10     scanf("%f", &b);
11
12     if (b!=0){
13         c = a/b;
14         printf("\n%f dividido por %f resulta em %f\n", a, b, c);
15     }
16     else{
17         printf("\nNão é possível realizar a divisão!!! ");
18         printf("Divisão por ZERO\n");
19     }
20
21     return 0;
22 }
23
```



Estrutura Condicional ou de Desvio

- A estrutura de desvio possibilita testar uma condição.
- **Se a condição for verdadeira**, um determinado conjunto de instruções é executado.
- **Se a condição for falsa**, outro conjunto de instruções é executado (ou o programa segue normalmente!)



Estrutura Condicional ou de Desvio

- Na linguagem C, um dos comandos para fazer o desvio (condição) é o comando **if** ou também conhecido como **if...else**



Estrutura Condicional ou de Desvio

- Exemplo 1: Verificar se o saldo de uma conta corrente é positivo.

```
#include <stdio.h>

int main(){
    float saldo;

    printf("Informe o saldo da conta corrente: ");
    scanf("%f", &saldo);

    if (saldo > 0){
        printf("Saldo Positivo\n");
    }else{
        printf("Saldo insuficiente\n");
    }
    return 0;
}
```

Estrutura Condicional ou de Desvio

- Exemplo 1: Verificar se o saldo de uma conta corrente é positivo.

```
#include <stdio.h>

int main(){
    float saldo;

    printf("Informe o saldo da conta co
    scanf("%f", &saldo);

    if (saldo > 0){
        printf("Saldo Positivo\n");
    }else{
        printf("Saldo insuficiente\n");
    }
    return 0;
}
```

■ Esta é a **condição** a ser testada

● Comando ou bloco de comandos a serem executados se a condição for **verdadeira**

● Comando ou bloco de comandos a serem executados se a condição for **falsa**



Estrutura Condicional ou de Desvio

- Exemplo 2:
Advinhar o número
gerado

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

int main(){
    int magic;
    int adv;

    srand(time(NULL));
    magic = rand()%10;

    printf("Advinhe o número mágico (de 0 a 10): ");
    scanf("%d", &adv);

    if (adv == magic)
        printf("*** Certo ***\n");
    else {
        printf("*** Errado ***\n");
        printf("O numero magico eh: %d\n", magic);
    }

    return 0;
}
```




Estrutura Condicional ou de Desvio

Observar:

- a indentação do código, pois facilita a leitura (legibilidade).
- se há apenas um comando no if, não há necessidade do uso do { }. Se existem vários comandos a serem executados na condição, é necessário o uso do delimitador de bloco { }

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

int main(){
    int magic;
    int adv;

    srand(time(NULL));
    magic = rand()%10;

    printf("Advinhe o número mágico (de 0 a 10): ");
    scanf("%d", &adv);

    if (adv == magic)
        printf("*** Certo ***\n");
    else {
        printf("*** Errado ***\n");
        printf("O numero magico eh: %d\n", magic);
    }

    return 0;
}
```



Estrutura Condicional ou de Desvio

- Exemplo 3:
Mostrar se um
número inteiro é par
ou ímpar

```
#include <stdio.h>

int main(){
    int numero;

    printf("Informe um número inteiro: ");
    scanf("%d", &numero);

    if (numero%2 ==0)
        printf("O numero eh par\n");
    else
        printf("O numero eh impar\n");

    return 0;
}
```



Estrutura Condicional ou de Desvio

- Desvio condicional simples, na linguagem C, tem o seguinte formato:

```
if condição {  
    //instruções          executadas          quando          a  
    //condição for verdadeira  
}
```

- Somente com opção para o caso da condição ser verdadeira (se for falsa, a execução continua normalmente).



Estrutura Condicional ou de Desvio

- Com conjunto de instruções para o caso da condição ser verdadeira e também para o caso da condição ser falsa:

```
if condição {  
    //instruções executadas quando a  
    //condição for verdadeira  
}else{  
    //instruções executadas quando a condição for falsa  
}
```



O que é a Condição?

- É o resultado de uma operação com operadores relacionais e lógicos!
- Esta operação resulta em VERDADEIRO ou FALSO.
- Em C, como não existe o tipo booleano:
 - Se o resultado for 0, é considerado FALSO.
 - Se for um número diferente de zero é considerado VERDADEIRO.



Operadores Relacionais

- São operadores que geram resultados lógicos (verdadeiro ou falso) pela comparação entre valores e/ou expressões.
- Os operadores relacionais têm a mesma prioridade.

Operador	Ação
>	Maior que
<	Menor que
>=	Maior ou igual a
<=	Menor ou igual a
==	Igual a
!=	Diferente de



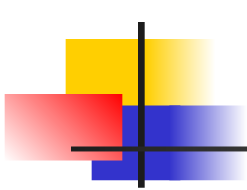
Operadores Relacionais (exemplos)

- Comparação entre valores
 - $100 > 5$ o resultado é verdadeiro
 - $200 \leq 30$ o resultado é falso
- Comparação entre variáveis
 - $a < b$ o resultado dependerá do valor das variáveis
- Comparação entre expressões
 - $(a + b) < (y * 10)$ o resultado dependerá do valor das variáveis nas expressões



Exercícios

- 1) Escreva um algoritmo para ler dois valores e informar qual é o maior valor lido.
- 2) Escreva um algoritmo para ler dois valores inteiros e escrever na tela a diferença do maior para o menor.
- 3) Calcule a média final de um aluno, considerando que esta é a média de três notas. Inclua a seguinte condição: se a média for maior ou igual a 6.0, o aluno estará Aprovado, caso contrário Reprovado.



Vamos praticar!

Lista de Exercícios 2

A blue L-shaped line in the bottom-right corner, consisting of a horizontal line segment followed by a vertical line segment extending upwards.