

ALGORITMOS E PROGRAMAÇÃO DE COMPUTADORES

AULA 2.1 – DESVIOS SIMPLES

AULA DE HOJE...

1. Desvios Simples - if

- ✓ Variável sinalizadora (*flag*)

2. Exercícios

DESVIOS SIMPLES

Considere os seguintes exemplos

·
·
·
<instrucao1>
<instrucao2>
<instrucao3>
<instrucao4>
<instrucao5>
<instrucao6>
·
·
·

Todas as instruções
são executadas

·
·
·
<instrucao1>
<instrucao2>
<condicao>
 <instrucao3>
<instrucao4>
<instrucao5>
<instrucao6>
·
·
·

Desvio no processamento;
A **<instrucao3>** está condicionada;
Só será executada se a condição for verdadeira;

DESVIOS SIMPLES

Relembrando alguns operadores...

Operadores relacionais

<

>

<=

>=

== (igual)

!= (diferente)

Operador de atribuição

=

DESVIOS SIMPLES

Estrutura

```
if(<condicao>)
```

```
    <instrucao1>; //instrução executada se e somente se a condição for satisfeita
```

```
    <instrucao2>;
```

```
    <instrucao3>;
```

```
    <instrucao4>;
```

DESVIOS SIMPLES

Estrutura

Exemplo em C

```
if(sexo == 'm')
```

```
if(<condicao>)
```

```
<instrucao1>;
```

```
<instrucao2>;
```

```
<instrucao3>;
```

```
<instrucao4>;
```

```
printf("Sexo masculino");
```

```
printf("Ola, este é um questionario");
```

```
printf("Informe sua idade");
```

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

Exemplo em C++

```
if(sexo == 'm')
```

```
cout<<"Sexo masculino";
```

```
cout<<""Ola, este é um questionario";
```

```
cin>>"Informe sua idade">>idade;
```

DESVIOS SIMPLES

O que está inconsistente nesse código?

Exemplo

```
if(idade > 60)
    printf("Senior");
printf("Prioridade no atendimento");
printf("Informe seu peso");
scanf("%f",&peso);
printf("Informe sua altura");
scanf("%f",&altura);
```

Exemplo

```
if(idade > 60)
    cout<<" Senior";
    cout<<" Prioridade no atendimento";
    cin>>"Informe seu peso">>peso;
    cin>>"Informe sua altura">>altura;
```

DESVIOS SIMPLES

Exemplo

```
if(idade > 60)
    printf("Senior");
printf("Prioridade no atendimento");
printf("Informe seu peso");
scanf("%f",&peso);
printf("Informe sua altura");
scanf("%f",&altura);
```

O que está inconsistente nesse código?

Instrução executada independentemente da condição

DESVIOS SIMPLES

Exemplo em C

```
if(idade > 60)
    printf("Senior");
printf("Prioridade no atendimento");
printf("Informe seu peso");
scanf("%f",&peso);
printf("Informe sua altura");
scanf("%f",&altura);
```

Uso de chaves {} para **associar duas ou mais instruções** à uma condição

```
if(idade > 60){
    printf("Senior");
    printf("Prioridade no atendimento");
}
printf("Informe seu peso");
scanf("%f",&peso);
printf("Informe sua altura");
scanf("%f",&altura);
```

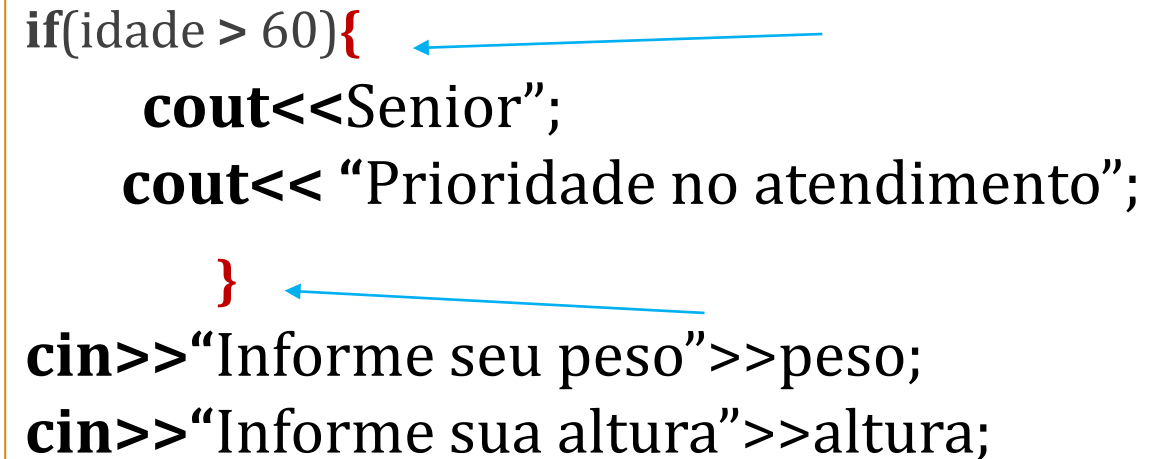
DESVIOS SIMPLES

Exemplo em C++

```
if(idade > 60)
    cout<<"Senior";
    cout<<"Prioridade no atendimento";
cin>>"Informe seu peso">>peso;
cin>>"Informe sua altura">>altura;
```

Uso de chaves {} para **associar duas ou mais instruções** à uma condição

```
if(idade > 60){
    cout<<"Senior";
    cout<<"Prioridade no atendimento";
}
cin>>"Informe seu peso">>peso;
cin>>"Informe sua altura">>altura;
```



VARIÁVEL SINALIZADORA

Variável sinalizadora (ou *flag*) em C++

- ✓ Sinaliza a realização de uma ou mais operações
- ✓ É um artifício utilizado para decidir sobre a execução de uma ou mais instruções
- ✓ É do tipo inteiro e geralmente assume valores binários

Se uma ou mais ações foram realizadas,
então não serão realizadas novamente

```
int flag = 0;
.
.
.
cout<<"Deseja realizar os cálculos agora?"<<endl;
cin>>flag;
if(flag) //(flag==1)
    cout<<"O resultado é:"<< X+Y-Z;
.
.
.
if(!flag) //(flag==0)
    cout<<"O resultado é: " << X+Y-Z;
.
.
.
```

VARIÁVEL SINALIZADORA

Variável sinalizadora (ou *flag*) em C

- ✓ Sinaliza a realização de uma ou mais operações
- ✓ É um artifício utilizado para decidir sobre a execução de uma ou mais instruções
- ✓ É do tipo inteiro e geralmente assume valores binários

Se uma ou mais ações foram realizadas,
então não serão realizadas novamente

```
int flag = 0;
.
.
.
printf("Deseja realizar os cálculos agora?");
scanf("%d",&flag);
if(flag) //(flag==1)
    printf("O resultado é: %f", X+Y-Z);
.
.
.
if(!flag) //(flag==0)
    printf("O resultado é: %f", X+Y-Z);
.
.
.
```

VARIÁVEL SINALIZADORA

Variável sinalizadora (ou *flag*)

- ✓ Sinaliza a realização de uma ou mais operações
- ✓ É um artifício utilizado para decidir sobre a execução de uma ou mais instruções
- ✓ É do tipo inteiro e geralmente assume valores binários

Se uma ou mais ações foram realizadas,
então outras poderão ou não ser realizadas

```
int flag = 0;
```

```
.  
.
```

```
printf("O código foi corretamente informado");  
flag = 1;
```

```
.  
.  
.
```

```
if(flag) //(flag==1)
```

```
printf("O valor atual é de: %f", saldo);
```

```
.  
.  
.
```

EXERCÍCIOS – LINGUAGEM C

A. Faça uma solução para calcular a idade de uma pessoa. Caso ela tenha 16 anos ou mais, exibir a seguinte mensagem: “usuário tem idade para votar”. Considere o ano atual para o cálculo

B. Faça uma solução para calcular a área de um quadrado. Exiba a área calculada. Se a área for menor ou igual a zero, exiba uma mensagem de erro

C. Faça uma solução para o usuário informar 5 valores. Para cada valor, exiba uma mensagem dizendo que o número é múltiplo de 3, caso ele realmente seja

EXERCÍCIOS – LINGUAGEM C

D. Fazer uma solução para o usuário informar 5 notas (elas têm os respectivos pesos: 2, 3, 3, 4 e 1). Mostrar a média ponderada. Se inferior a 5, exibir mensagem de reprovado

E. Fazer uma solução para o usuário informar seu salário atual. Se o salário for menor que 2000, exibir aumento de 10%. Mostrar o salário (possivelmente já reajustado)

EXERCÍCIOS – LINGUAGEM C

F. Fazer uma solução para fazer a seguinte verificação:

Se casado, informar dia/mes/ano do casamento. Para Qualquer caso (casado ou não), solicitar número de filhos, anos de empresa e pretensão salarial.

Exibir uma mensagem com todas as informações fornecidas pelo usuário

G. Proponha e faça uma solução para demonstrar o uso de variável sinalizadora (flag). Entregar!