

LINGUAGEM C – UMA INTRODUÇÃO

AULA 6 – Estruturas de decisão

As estruturas de decisão estão presentes em qualquer linguagem de programação. Com elas podemos realizar testes ao longo de nosso programa e decidir se o fluxo do mesmo deve ser desviado para outra direção ou não.

1 – O comando *if()*

If quer dizer “se” e é a estrutura de decisão mais usada em Linguagem C. Sua forma mais simples de uso é:

```
if ( "expressão de teste" ) {  
    instruções;  
    instruções;  
}
```

Se a “expressão de teste” for verdadeira (algo diferente de 0), o comando *if* executa as instruções entre { e }. Caso contrário, o programa salta para a linha imediatamente após }. As “expressões de teste” normalmente são realizadas com operadores relacionais (==, !=, >=, <=, >, <).

Problema: Escreva um programa em Linguagem C que teste se um valor digitado pelo usuário é menor do que 10.

Um algoritmo para resolver o problema proposto poderia ser:

- Receba o número “n”
- Se $n < 10$
 - Escreva “Ok.. o número é menor do que 10”
- Se $n \geq 10$
 - Escreva “O número não é menor do que 10”

Implementando em Linguagem C:

Exemplo 34

```
/*  
Primeiro exemplo com IF  
Linguagem C - Uma introdução  
--  
Diego M. Rodrigues  
*/  
#include <stdio.h>  
  
int main() {  
    //Declarando as variáveis  
    int n=0;  
  
    //Recebendo a variável n  
    printf("Digite um numero inteiro menor do que 10:\n");
```

```

scanf("%d", &n);

//Testando se n é menor do que 10
if ( n < 10 ) {
    printf("\nOk... o numero e menor do que 10\n");
}

if ( n >= 10 ) {
    printf("\nO numero NAO e menor do que 10!!!\n");
}

//Esperando o <ENTER> para sair
getch();

return 0;
}

```

Exemplos de execução do programa anterior:

```

C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo34.exe
Digite um numero inteiro menor do que 10:
8
Ok... o numero e menor do que 10

```

```

C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo34.exe
Digite um numero inteiro menor do que 10:
25
O numero NAO e menor do que 10!!!

```

2. O comando else

Podemos pensar no comando *else* como um complemento (ou uma negação) do comando *if*. A forma um pouco mais completa da estrutura *if-else* tem a seguinte forma geral:

```

if ( "expressão de teste" ) {
    instruções1;
    instruções2;
} else {
    instruções3;
    instruções4;
}

```

A “*expressão de teste*” será avaliada. Se ela for verdadeira (diferente de zero) o primeiro bloco de instruções será executado (instruções1; instruções2;). Caso a expressão de teste seja falsa (ou igual a zero) o segundo bloco de instruções será executado (instruções3; instruções4;). É importante ressaltar que quando usamos a estrutura *if-else*, estamos garantindo que um dos dois blocos de instruções será executado. Nunca serão executados os dois blocos ou nenhum deles.

Problema: Uma grande dificuldade em viagens internacionais é a conversão dos valores de temperatura de °C para °F. Faça um programa em Linguagem C que receba

uma temperatura em °F e calcule seu equivalente em °C. O programa deve exibir a mensagem “Levar roupas de frio.” quando a temperatura em °C for menor ou igual à 20 e mostrar a mensagem “Não levar roupas de frio.” caso contrário.

Um algoritmo para resolver o problema proposto poderia ser:

- Receba a temperatura °F em “f”
- Calcule a conversão para °C com a fórmula $c = ((f-32)*5)/9$
- Se $c \leq 20$
 - Escreva a temperatura em °C na tela
 - Escreva “Levar roupas de frio.”

Caso contrário

- Escreva a temperatura em °C na tela
- Escreva “Não levar roupas de frio.”

Implementando em Linguagem C:

Exemplo 35

```
/*
Exemplo com IF-ELSE - Conversão Fahrenheit-Centígrados
Linguagem C - Uma introdução
--
Diego M. Rodrigues
*/
#include <stdio.h>

int main() {
    //Declarando as variáveis
    float f=0, c=0;

    //Recebendo a temperatura f
    printf("Digite a temperatura em Fahrenheit:\n");
    scanf("%f", &f);

    //Calculando a conversão
    c=( (f-32)*5 )/9;

    //Testando se n é menor do que 10
    if ( c <= 20 ) {
        printf("\nTemperatura: %4.1f graus C\n", c);
        printf("\nLevar roupas de frio.\n");
    }
    else {
        printf("\nTemperatura: %4.1f graus C\n", c);
        printf("\nNão levar roupas de frio.\n");
    }

    //Esperando o <ENTER> para sair
    getch();

    return 0;
}
```

Exemplos de execução do programa anterior:

```
C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo35.exe
Digite a temperatura em Fahrenheit:
60
Temperatura: 15.6 graus C
Levar roupas de frio.
```

```
C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo35.exe
Digite a temperatura em Fahrenheit:
95
Temperatura: 35.0 graus C
Nao levar roupas de frio.
```

Problema: Escreva um programa em Linguagem C que calcule a média de um aluno, com base em duas provas (mensal e bimestral) e mostre a mensagem “Aluno Aprovado” se a média for maior ou igual do que 5 e “Aluno Reprovado” caso contrário.

Um algoritmo para resolver o problema proposto poderia ser:

- Receba a nota da prova mensal
- Receba a nota da prova bimestral
- Calcule a média com a fórmula $media = (mensal + bimestral) / 2$
- Se $media \geq 5$
 - Escreva “Aluno aprovado!”
- Caso contrário
 - Escreva “Aluno Reprovado!”
- Exiba a média na tela

O programa que realiza a tarefa acima poderia ser escrito em uma pseudo linguagem da seguinte forma:

Implementando em Linguagem C:

```
Exemplo 36
/*
Exemplo com IF-ELSE - Cálculo de média
Linguagem C - Uma introdução
--
Diego M. Rodrigues
*/
#include <stdio.h>

int main() {

    //Declarando as variáveis
    float mensal=0;
    float bimestral=0;
    float media=0;

    //Recebendo a nota mensal
    printf("\nDigite a nota mensal: ");
    scanf("%f",&mensal);
```

```

//Recebendo a nota bimestral
printf("\nDigite a nota bimestral: ");
scanf("%f",&bimestral);

//Calculando a média
media = (mensal+bimestral)/2;

//Mostrando mensagens
if ( media >= 5 ) {
    printf("\n\nAluno aprovado!");
}
else {
    printf("\n\nAluno reprovado!");
}

//Mostrando a média
printf("\nA media do aluno e: %4.1f",media);

//Esperando o <ENTER> para sair
getch();
return 0;
}

```

Exemplos de execução do programa acima:

```

C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo36.exe
Digite a nota mensal: 10
Digite a nota bimestral: 8

Aluno aprovado!
A media do aluno e: 9.0_

```

```

C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo36.exe
Digite a nota mensal: 5.5
Digite a nota bimestral: 3.8

Aluno reprovado!
A media do aluno e: 4.7

```

3 – O else if

Em todos os testes realizados nos exemplos anteriores nossos programas escolhiam uma dentre duas opções (se... caso contrário...). Em muitos casos precisamos realizar testes mais elaborados, com mais de duas opções (se... se não, se... caso contrário...), para isso podemos usar o *else if*.

Dessa forma, chegamos à forma completa, *if - else if - else*:

```

if ( "expressão de teste 1" ) {
    instruções1;
    instruções2;
}
else if ( "expressão de teste 2" ) {
    instruções3;
    instruções4;
}
else {
    Instruções5;
    Instruções6;
}

```

A “*expressão de teste 1*” será a primeira avaliada. Se ela for verdadeira (diferente de zero) o primeiro bloco de instruções será executado (instuções1; instruções2;). Caso a “*expressão de teste 1*” seja falsa (ou igual a zero), a “*expressão de teste 2*” será avaliada. Caso ela seja verdadeira, o segundo bloco de instruções será executado (instuções3; instruções4;). Caso tanto a “*expressão de teste 1*” quanto a “*expressão de teste 2*” sejam falsas, o terceiro bloco de instruções será executado (instuções5; instruções6;).

Problema: Escreva um programa em Linguagem C que receba um número n e imprima na tela:

- “O número é positivo”, se o usuário digitar um número positivo
- “O número é negativo”, se o usuário digitar um número negativo
- “Você digitou zero”, se o usuário digitar o número zero

Um algoritmo para resolver o problema proposto poderia ser:

- Receba o número n
- Se $n > 0$
 - Escreva “Número positivo!”
- Se não, se $n < 0$
 - Escreva “Número negativo!”
- Caso contrário
 - Escreva “Você digitou zero!”

Exemplo 37

```

/*
Exemplo com IF - ELSE IF - ELSE - Numeros positivos ou negativos
Linguagem C - Uma introdução
--
Diego M. Rodrigues
*/
#include <stdio.h>

int main() {

    //Declarando as variáveis
    int n=0;

    //Recebendo o número
    printf("\nDigite um numero: ");
    scanf ("%d",&n);

    //Mostando mensagens

```

```

if ( n > 0 ) {
    printf("\n\nO numero e positivo!");
}
else if ( n < 0 ) {
    printf("\n\nO numero e negativo!");
}
else {
    printf("\n\nVoce digitou zero!");
}

//Esperando o <ENTER> para sair
getch();
return 0;
}

```

Exemplos de execução do programa acima:

The screenshot shows a command prompt window titled "C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo37.exe". The prompt asks "Digite um numero:" and the user has entered "29". The program's output is "O numero e positivo!".

The screenshot shows a command prompt window titled "C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo37.exe". The prompt asks "Digite um numero:" and the user has entered "-95". The program's output is "O numero e negativo!".

The screenshot shows a command prompt window titled "C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo37.exe". The prompt asks "Digite um numero:" and the user has entered "0". The program's output is "Voce digitou zero!".

Problema: Desenvolva uma calculadora simples em Liguagem C. Essa calculadora deve ser capaz de realizar as 4 operações básicas (+, -, *, /) sobre dois números quaisquer.

Observações:

- Para ler a operação que será realizada (+, -, *, /), use o %c dentro do `scanf()`.
- Como a operação será lida com %c (como caractere), você deve usar aspas simples na “expressão de teste”, coisas do tipo: `if (operador == '+')`
- Você pode ler mais de uma variável dentro do mesmo `scanf()`, desde que elas fiquem separadas por espaços. Ex: `scanf("%d %d", &num1, &num2)`

Um algoritmo para resolver o problema proposto poderia ser:

- Escreva uma mensagem de boas vindas
- Receba o operando 1
- Receba o operador

- Receba o operando 2
- Se operador == '+'
 - Calcular resultado = operando1 + operando2
- Se não, se operador == '-'
 - Calcular resultado = operando1 - operando2
- Se não, se operador == '*'
 - Calcular resultado = operando1 * operando2
- Se não, se operador == '/'
 - Calcular resultado = operando1 / operando2
- Escreva o resultado na tela

Exemplo 38

```

/*
Exemplo com IF - ELSE IF - ELSE - Numeros positivos ou negativos
Linguagem C - Uma introdução
--
Diego M. Rodrigues
*/
#include <stdio.h>

int main() {
    //Declarando as variáveis
    float num1=0, num2=0, resultado=0;
    char operador;

    //Mensagem de boas vindas
    printf("Bem vindo a super calculadora!");
    printf("\nVoce deve digitar expressoes da forma: num1 op num2");
    printf("\nExemplos:");
    printf("\n2 + 5");
    printf("\n8 * 3");
    printf("\nNao esqueca dos espacos!");

    //Recebendo o número
    printf("\n\nDigite uma expressao no formato: num1 op num2: ");
    scanf("%f %c %f", &num1, &operador, &num2);

    //Mostrando mensagens
    if ( operador == '+' ) {
        resultado = num1 + num2;
    }
    else if ( operador == '-' ) {
        resultado = num1 - num2;
    }
    else if ( operador == '*' ) {
        resultado = num1 * num2;
    }
    else if ( operador == '/' ) {
        resultado = num1 / num2;
    }
    }

    //Mostrando o resultado
    printf("\n\nO resultado = %4.1f",resultado);

    //Esperando o <ENTER> para sair
    getch();
    return 0;
}

```

Exemplos de execução do programa anterior:

```
C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo38.exe
 Bem vindo a super calculadora!
 Uoce deve digitar expressoes da forma: num1 op num2
 Exemplos:
 2 + 5
 8 * 3
 Nao esqueca dos espacos!

 Digite uma expressao no formato: num1 op num2: 5 + 12

 Resultado = 17.0_
```

```
C:\Documents and Settings\Diego\Meus documentos\Exemplos\exemplo38.exe
 Bem vindo a super calculadora!
 Uoce deve digitar expressoes da forma: num1 op num2
 Exemplos:
 2 + 5
 8 * 3
 Nao esqueca dos espacos!

 Digite uma expressao no formato: num1 op num2: 5 * 3

 Resultado = 15.0_
```

Observações finais sobre esse exemplo:

A expressão matemática foi lida na linha:

```
scanf("%f %c %f", &num1, &operador, &num2);
```

Como já foi dito, podemos ler mais de uma variável do teclado dentro de um mesmo *scanf()*, desde que tudo fique separado por espaços. No *scanf()* acima o primeiro *%f* indica que a primeira variável (*num1*) irá receber um valor do tipo *float*. O *%c* indica que a segunda variável (*operador*) irá receber um *char* e finalmente o último *%f* indica que a última variável (*num3*) irá receber um valor *float*.

```
scanf("%f %c %f", &num1, &operador, &num2);
```

Vamos dar outro exemplo de *scanf()* composto, mas dessa vez com quatro variáveis: idade(*int*), peso(*float*), sexo(*char*) e saldo(*double*):

```
scanf("%d %f %c %g", &idade, &peso, &sexo, &saldo);
```

Outra coisa que deve ser ressaltada é que sempre que usamos **caracteres isolados no programa, estes devem estar em ASPAS SIMPLES**. Observe como exemplo o pequeno trecho de código abaixo que recebe um *char* dentro de uma variável *sexo* para testar se o valor digitado foi *H* ou *M*.

```
char sexo;
printf("\nQual o sexo (H/M)? ");
scanf("%c", &sexo);
if ( sexo == 'H' ) {
    printf("Homem!");
} else if ( sexo == 'M' ) {
    printf("Mulher!");
} else {
    printf("Opção inválida!");
}
```