

PROGRAMAÇÃO IMPERATIVA

Prof. Dr. Edson J. R. Justino

Prof. Dr. Alcides Calsavara

Semana 3

Pontifícia Universidade Católica do Paraná – PUCPR
Escola Politécnica

Operadores

Relacionais

- Quando queremos estabelecer uma comparação entre dois valores para tomar uma decisão, utilizamos as operações relacionais.

> maior
< menor
>= maior ou igual
<= menor ou igual
== igual
!= diferente (não igual)

OBS: Quando elaboramos programa, devemos tomar um cuidado especial para não confundirmos as formas:

a == b a = b

```
/******  
Exemplo de Operadores  
Relacionais  
*****/  
#include <stdio.h>  
  
int main ( ) {  
  
    int a, b;  
  
    b = 10;  
    a = 2;  
    if(a == b)  
        puts("iguais");  
    else  
        puts("diferentes");  
    return 0;  
}
```

Operadores

● Lógicos

- Em C temos três operadores lógicos, que utilizam normalmente operações relacionais como operadores.

&& (E)

|| (OU)

! (NÃO)

```
/******  
Exemplo de Operadores Relacionais  
******/  
#include <stdio.h>  
  
int main ( ) {  
  
    int a, b;  
  
    a = 2;  
    b = 3;  
    if ( (a == 2) && (b == 3) )  
        printf("a = 2 e b = 3\n");  
    if ( (a == 2) || (b == 3) )  
        printf("a = 2 ou b = 3 ou ambos\n");  
    if (a != 3)  
        puts("a é diferente de 3\n");  
    return 0;  
}
```

Comando if Simples

- Em C, a avaliação de expressão é *semibooleana*, quer dizer, uma expressão será “verdadeira” se, quando avaliada, produza um resultado diferente de zero, e “falso” se seu resultado for igual a zero.

- Sintaxe:

```
if(expressão)
{
    comando;
}
```

OBS: Quando existir apenas um comando, as chaves são opcionais

Comando if Composto

- Sintaxe:

```
if(expressão) {  
    comando;  
}  
else {  
    comando;  
}
```

OBS: Quando existir apenas um comando em uma das condições, as chaves são opcionais

```
/******
```

Exemplo do comando if

```
******/
```

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

```
int main( ){
```

```
    int a, b, z;
```

```
    b = 2, a = 10;
```

```
    if (a > 0)
```

```
        printf ("valor positivo\n");
```

```
    if (a < b){
```

```
        z = a;
```

```
        printf("A e menor que B\n Z = %d", z);
```

```
    }
```

```
    else {
```

```
        z = b;
```

```
        printf("B e menor que A\n Z = %d", z);
```

```
    }
```

```
    return(0);
```

```
}
```

Comando while

- O comando *while* é utilizado para a execução de um bloco de comandos enquanto uma condição lógica for “verdadeira”.

- Sintaxe:

```
while (expressão) {  
    comandos;  
}
```

OBS: Quando existir apenas um comando, as chaves são opcionais

```
/******  
Exemplo de Comando While  
*****/  
#include <stdio.h>  
  
int main()  
{  
    int    a = 10, b = 0;  
  
    while ((a != 0) || (b != 10)){  
        printf("A = %d  B = %d\n",a, b);  
        a--;  
        b++;  
    }  
    return(0);  
}
```

Exercícios Propostos

- 1) Elabore um pgm. em C que implemente o código em Python ao lado.

```
# Python
PI = 3.1416
R = 0
while R <= 6:
    VOLUME = 4/3 * PI * (R**3)
    print(R, VOLUME)
    R = R + 2
```


Exercícios Propostos

- 2) Elabore um pgm. em C que implemente o código em Python ao lado.

```
# Python
NÚMERO = 1352
D4 = NÚMERO % 10
D3 = (NÚMERO // 10) % 10
D2 = (NÚMERO // 100) % 10
D1 = (NÚMERO // 1000) % 10
print (D4, D3, D2, D1)
```

Exercícios Propostos

- 3) Elabore um pgm. em C que implemente o código em Python ao lado.
- Após a implementação, teste com os seguintes valores:
 - a) NUM = 10
 - b) NUM = 0
 - c) NUM = -4

OBS:

Para atribuir um valor cadeia de caracteres a uma variável, deve ser usada a função strcpy()

Ex:

```
#include <string.h>
```

```
:
```

```
strcpy (QUALQUER, "Numero Positivo");
```

```
# Python
NUM = int(input("Valor: ")) # um dos valores ao lado
if NUM > 0:
    QUALQUER = "Número positivo"
elif NUM < 0:
    QUALQUER = "Número negativo"
else:
    QUALQUER = "Zero"
print (QUALQUER)
```

Exercícios Propostos

- 4) Elabore um pgm. em C que implemente o código em Python ao lado.

```
# Python
N = int (input("Entre com o valor: "))
X = N % 2
if X == 0:
    VAL = "Número Par"
else:
    VAL = "Número Impar"
print (" O número é:", VAL)
```

Exercícios Propostos

- 5) Elabore um pgm. em C que implemente o código em Python ao lado.

```
# Python
# recebe o valor lido
VALOR = int (input("Entre com o valor: "))

# contém, a cada instante, o menor valor lido
MENOR = VALOR
while VALOR != -1: #condição de parada
    if VALOR < MENOR:
        MENOR = VALOR
    VALOR = int (input("Entre com o valor: "))
print (" O menor valor lido é:", MENOR)
```

Exercícios Propostos

- 1) Elaborar um pgm em C para fazer a soma de dois valores inteiros e positivos lidos do teclado.
- 2) Elaborar um pgm em C para calcular a média de um conjunto de valores inteiros e positivos lidos do teclado. Para concluir a leitura o valor digitado deve ser -1.
- 3) Elaborar um pgm em C para calcular as raízes de uma equação do 2o. Grau, sendo que os valores dos coeficientes A, B e C serão digitados.
- 4) Sendo $H = 1 + 1/2 + 1/3 + 1/4 + \dots + 1/N$, elaborar um pgm em C para gerar o número H. O valor de N deverá ser digitado.
- 5) Escreva um pgm em C para calcular o fatorial de um número N digitado
- 6) Fazer um pgm em C para calcular o valor da série S. O valor de N deve ser digitado
$$S = 1/N + 2/N-1 + 3/N-2 + \dots + N-1/2 + N/1$$
- 7) Elaborar um pgm. em C para calcular a série de Fibonacci para os 20 primeiros termos.
 - 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ... etc.
- 8) Elaborar um pgm. em C para imprimir os 30 primeiros números pares.