



# Engenharia da Computação

**Disciplina:** DCExt Programação Imperativa

**Aula:** Conceitos Fundamentais (parte 2)

Prof. Dr. Hemir da C. Santiago  
hcs2@poli.br

## Aula 03.2:



- Operador de atribuição
- Operadores de comparação e relacionais
- Operadores lógicos

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

```
int main () {  
    int a, b, c, y;  
    float d;  
    a = 2;  
    b = 5;  
    c = 10;  
    d = 2.0;  
    c = 5;
```

```
    printf("a = %d\n", a);  
    printf("b = %d\n", b);  
    printf("c = %d\n", c);  
    printf("d = %.1f\n", d);  
    printf("c/a = %d\n", c/a);  
    printf("c/d = %d\n", c/d);  
    printf("c/d = %.1f\n", c/d);  
    y = a + b + c + d;  
    printf("y = %d\n", y);
```

### Saídas:

```
a = 2  
b = 5  
c = 5  
d = 2.0  
c/a = 2  
c/d = 0  
c/d = 2.5  
y = 14
```

```
}
```

## Operador de atribuição

Os operadores de comparação e relacionais, apresentados na tabela abaixo, podem ser empregados para fazer comparação entre valores, entre variáveis (que armazenam dados ou ainda entre expressões).

Operador	Operação
==	Igual a
!=	Não igual a (Diferente de)
>	Maior do que
<	Menor do que
>=	Maior do que ou igual a
<=	Menor do que ou igual a

## Operadores de comparação e relacionais

Quando uma operação é realizada, o resultado é **verdadeiro** ou **falso**, isto é, um **valor** ou **estado lógico** (também chamado de **booleano**). Na linguagem C: **verdadeiro = 1** e **falso = 0**.

Por exemplo, as comparações seguintes resultam: *verdadeiro(1)*, *falso(0)*, *verdadeiro(1)*, respectivamente.

$(2 == 2)$

$(2 != 2)$

$(3 >= 2)$

# Operadores de comparação e relacionais

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

```
int main () {  
    int a, b, c;  
    a = 4;  
    b = 2;  
    c = 3;
```

**Saídas:**

```
printf("%d\n", a == b);    0  
printf("%d\n", a != b);    1  
printf("%d\n", a == (b + 2)); 1  
printf("%d\n", a > c);      1  
printf("%d\n", b < c);      1  
printf("%d\n", (a + b) >= (a + c)); 0  
printf("%d\n", (a - b) >= (a - c)); 1
```

```
}
```

## Operadores de comparação e relacionais

Há três operadores lógicos, apresentados na tabela abaixo, que podem ser ***empregados para avaliar uma expressão lógica***.

Operador	Operação
!	Negação de
&	E lógico
	OU lógico

Por exemplo, considere a instrução:

***printf(“%d”, !2 == 2);***

A expressão (2 == 2) é avaliada como *verdadeira*. Contudo, como o operador de negação (!) atua sobre ela, resulta ***false*** que é exibido como ***0***.

## Operadores lógicos

O **operador lógico and ( & )** é utilizado para avaliar dois operandos (ou expressões) resultando verdadeiro se os dois operandos são verdadeiros; caso contrário, resulta falso, como ilustrado na tabela abaixo.

Operando a	Operando b	Operação a and b
Verdadeiro	Verdadeiro	Verdadeiro
Verdadeiro	Falso	Falso
Falso	Verdadeiro	Falso
Falso	Falso	Falso

## Operadores lógicos



O **operador lógico or ( / )** é utilizado para avaliar dois operandos (ou expressões) resultando verdadeiro se, pelo menos, um dos dois operandos é verdadeiro. O resultado é falso se ambos os operandos são falsos, como ilustrado na tabela abaixo.

Operando a	Operando b	Operação a or b
Verdadeiro	Verdadeiro	Verdadeiro
Verdadeiro	Falso	Verdadeiro
Falso	Verdadeiro	Verdadeiro
Falso	Falso	Falso

## Operadores lógicos

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

```
int main () {  
    int a, b, c;  
    a = 4;  
    b = 2;  
    c = 3;
```

**Saídas:**

```
printf("%d\n", (2 == 2) & (a > b));    1  
printf("%d\n", (a != b) | (4 < 7));    1  
printf("%d\n", (a == c) & (b >= 2));    0  
printf("%d\n", !(a > c) & (b <= c));    0  
printf("%d\n", (b < c) | !(1));        1  
printf("%d\n", !(a != 4) & !(0));      1
```

```
}
```

## Operadores lógicos

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

```
int main () {  
    int a, b, c;  
    a = 4;  
    b = 2;  
    c = 3;
```

**Saídas:**

```
printf("%d\n", ++a);      5  
printf("%d\n", --a);      4  
c++;  
printf("%d\n", --a == c); 0  
printf("%d\n", a >= c++); 0  
c--;  
printf("%d\n", b <= --c);  1  
printf("%d\n", ++a != 4); 0  
}
```

## Operadores lógicos

1) Elaborar um programa para obter o resultado das seguintes expressões:

$(2 == 2)$

$(2 != 2)$

$(2 > 2)$

$(2 >= 2)$

$(2 < 2)$

$(2 <= 2)$

## Exercícios

	DATA	AULA
1	22/08/2024	<b>Apresentação da disciplina   Introdução à Programação Imperativa</b>
2	29/08/2024	<b>Introdução à Linguagem de Programação C</b>
3	05/09/2024	<b>Conceitos Fundamentais</b>
4	12/09/2024	<b>Tipos de Dados Especiais em C</b>
5	19/09/2024	<b>Estruturas Condicionais e de Repetição</b>
6	26/09/2024	<b>Pré-processamento</b>
7	03/10/2024	<b>Registros/Estruturas de Dados</b>
8	10/10/2024	<b>Ponteiros</b>
9	17/10/2024	<b>1º Exercício Escolar</b>

# Plano de Aulas

	DATA	AULA
10	24/10/2024	Arquivos
11	31/10/2024	Acompanhamento de projetos
12	07/11/2024	Acompanhamento de projetos
13	14/11/2024	Acompanhamento de projetos
14	21/11/2024	Acompanhamento de projetos
15	28/11/2024	Apresentação parcial
16	05/12/2024	Apresentação de projetos
17	12/12/2024	Avaliação Final

# Plano de Aulas