

# Operadores e Expressões

Prof. Gabriel Barbosa da Fonseca

Adaptado de Profa. Rosilane Mota

## Operadores Matemáticos

---

+	2 + 3 x + 2	Soma
-	7 - 8 y - 5	Subtração
*	2 * 6 3 * z	Multiplicação
/	3.0 / 2 (1.5) 3 / 2 (1) 3 / 2.0 (1.5)	Divisão
%	3 % 2 (1)	Resto da divisão

---

## Operadores Matemáticos de Atribuição

---

`+=` `-=` `*=` `/=` `%=`

`x += y` equivale a `x = x + y`

`x += 8` equivale a `x = x + 8`

`++` `--`

`x++` equivale a `x = x + 1`

`y--` equivale a `y = y - 1`

---

## Operadores Relacionais (Binários)

---

==	x == (y + 2)	Igualdade
	z == 3	
!=	x != (y + 2)	Diferença
	z != 3	
<	x < y	Menor
<=	x <= y	Menor ou igual
>	x > y	Maior
>=	x >= y	Maior ou igual

---

## Operadores Lógicos

---

!	!achou	Negação
	!true (false)	
	!false (true)	

### **E** lógico (And)

Op1 Op2 Op1 **&&** Op2

V	V	V
V	F	F
F	V	F
F	F	F

### **Ou** lógico (Or)

Op1 Op2 Op1 **||** Op2

V	V	V
V	F	V
F	V	V
F	F	F

---

## Exemplo

---

`x = 2; y = 3; z = 4; achou = false;`

`(x < y + 1) && (!achou) && (z == x + 3)`

`(2 < 3 + 1) && (!false) && (4 == 2 + 3)`

`(2 < 4) && true && (4 == 5)`

`true && true && false`

`true && false`

`false`

---

## Funções Matemáticas em C

---

Incluir a biblioteca **math.h**

O resultado de cada uma delas é do tipo **double**

---

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

```
#include <math.h>
```

```
#define PI 3.14159265
```

```
int main ()
```

```
{
```

```
    double x, ret, val;
```

```
    x = 60.0;
```

```
    val = PI / 180.0;
```

```
    ret = cos( x*val );
```

```
    printf("Cosseno de %lf é %lf graus\n", x, ret);
```

```
    return(0);
```

```
}
```

## Funções Matemáticas em C

---

Função	Descrição do comando
<code>floor( )</code>	arredonda para baixo
<code>ceil( )</code>	arredonda para cima
<code>sqrt( )</code>	Calcula raiz quadrada
<code>pow(variável, expoente)</code>	potenciação
<code>sin( )</code>	seno
<code>cos( )</code>	cosseno
<code>tan( )</code>	Tangente
<code>log( )</code>	logaritmo natural
<code>log10( )</code>	logaritmo base 10

---

## Funções Matemáticas em C

---

O sorteio de números é muito utilizado. Pode-se utilizar a hora do sistema como base para geração dos números aleatórios com a função **srand** e **time**, além de definir um intervalo (mínimo e máximo) dos valores que serão sorteados por uma expressão utilizada com a função **rand**.

---

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

```
int main () {

    // Inicializa gerador de números
    srand((unsigned) time(NULL));

    // Sorteia número entre 40 e 149
    printf("%d\n", (rand() % (150-40))+40);

    return(0);
}
```

# Operadores e Expressões

Prof. Gabriel Barbosa da Fonseca

Adaptado de Profa. Rosilane Mota