

# Algoritmos - AULA 3

---

## Sumário

1	Operadores de Atribuição Aritmética e Operadores Incrementais .....	1
2	Teste de Mesa.....	1
2.1	Como fazer um teste de mesa? .....	2
3	Funções matemáticas.....	3
4	Exemplo completo .....	3
5	Exercícios .....	3

---

## 1 Operadores de Atribuição Aritmética e Operadores Incrementais

Algumas vezes podemos usar uma forma reduzida para realizar instruções mais longas de uma forma mais curta, a saber:

Exemplo de uma Forma Normal	Exemplo de uma Forma Reduzida	Exemplo			
		Valor de X antes da instrução		Valor de X depois da instrução	
x = x + 1;	x++;	2		3	
x = x - 1;	x--;	2		1	
x = x + 5;	x += 5;	2		7	
x = x - 5;	x -= 5;	2		-3	
x = x * 3;	x *= 3;	2		6	
x = x / 3;	x /= 3;	6		2	
Exemplo de uma Forma Normal	Exemplo de uma Forma Reduzida	Exemplo			
		Valor de x antes da instrução	Valor de y antes da instrução	Valor de x depois da instrução	Valor de y depois da instrução
x = x + 1; y = x;	y = ++x;	3	7	4	4
x = x - 1; y = x;	y = --x;	3	7	2	2
y = x; x = x + 1;	y = x++;	3	7	4	3
y = x; x = x - 1;	y = x--;	3	7	2	3

## 2 Teste de Mesa

Na fase de desenvolvimento de um programa, muitas vezes precisamos testar se as funcionalidades implementadas em um algoritmo estão corretas. Para isso, é necessário verificar o conteúdo das variáveis passo a passo, o que chamamos de teste de mesa.

Ou seja, precisamos seguir passo a passo as instruções de um algoritmo, de maneira precisa, para verificar se o algoritmo foi implementado da forma correta, a partir dos resultados gerados e dos valores parciais de cada variável.

Com o teste de mesa é possível:

- Visualizar o comportamento de todo o processo;
- Identificar e corrigir eventuais erros (“bugs”), se existirem;
  - Erros de sintaxe: quando as instruções do programa não são escritas de acordo com a sintaxe da linguagem usada;
  - Erros lógicos: quando o código escrito pelo programador não gera o resultado desejado pelo mesmo.

## 2.1 Como fazer um teste de mesa?

Usando o caderno ou algum bloco de anotação, faça o seguinte:

- Crie uma tabela e no cabeçalho coloque as variáveis declaradas. Crie uma coluna para cada variável declarada;
- Crie um espaço para anotar tudo o que deve aparecer na tela do computador;
- Execute o algoritmo, manual e visualmente, linha a linha, anotando nas colunas apropriadas cada mudança de valor das variáveis, e no lugar você criou tudo o que foi escrito na tela;
- Preste atenção nas estruturas condicionais (porque pode haver instruções que não serão executadas) e nas estruturas de repetição (porque pode haver trechos de instruções que devem ser executados mais de uma vez).
- Siga a execução até chegar ao final do algoritmo
- Execute o programa no computador e compare o resultado do seu teste de mesa com o resultado obtido no computador.

### Exemplo

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int a,b,c,d;
    a=0; b=0; c=0; d=0;
    a=8;
    b=12;
    d=a+a;
    c=a+b;
    a=c%b;
    d=d+1;
    a=a+c;
    c=d-b;
    printf("a:%d b:%d c:%d d:%d\n",a,b,c,d);
    system("pause");
    return 0;
}
```

Teste de Mesa

a	b	c	d
0	0	0	0
8	12	20	16
8		5	17
28			

Na tela:

a:28 b:12 c:5 d:17

### 3 Funções matemáticas

O `math.h` é um arquivo cabeçalho que fornece protótipos para funções, macros e definição de tipos da biblioteca padrão da linguagem de programação C para funções matemáticas básicas. São disponibilizadas, das quais podemos citar:

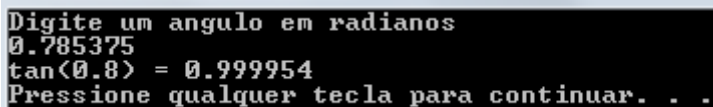
Trigonométricas

- `sin(ângulo)`: retorna o valor do seno. Recebe como argumento o valor dos graus em radianos.
- `cos(ângulo)`: retorna o valor do co-seno. Recebe como argumento o valor dos graus em radianos.
- `tan(ângulo)`: retorna o valor da tangente. Recebe como argumento o valor dos graus em radianos.
- `pow(base,expoente)`: retorna o valor da base elevada ao expoente. Recebe dois argumentos, o primeiro é a base e o segundo o expoente.
- `sqrt(valor)`: retorna o valor da raiz quadrada. Recebe como argumento um número do qual ele deve extrair a raiz.

### 4 Exemplo completo

Entrar com o valor de 45 graus, em radianos, e mostrar o valor da tangente.

```
#include<stdio.h>
#include<stdlib.h>
#include<math.h>
int main()
{
    float angulo;
    printf("Digite um angulo em radianos\n");
    scanf("%f",&angulo);
    printf("tan(%.1f) = %f\n",angulo,tan(angulo));
    system("pause");
    return 0;
}
```



```
Digite um angulo em radianos
0.785375
tan(0.8) = 0.999954
Pressione qualquer tecla para continuar. . .
```

### 5 Exercícios

- 1 Faça o teste de mesa e mostre o último valor armazenado em cada uma das variáveis.

```
#include <stdio.h>
#include <stdlib.h>
int main (void)
{
    float x,y=2.5;
    int a,b=1,c;
    x=++y;
    x*=2;
    a=b+1;
    a++;
```

```
c=2*a+b;
a=c--;
b=a++ + --c;
printf("x =%.1f\nty=%.1f\na=%i\nb=%i\nc=%i\n",x,y,a,b,c);
system("pause");
return 0;
}
```

- 2 Leia dois valores para as variáveis A e B e efetua a troca dos valores de forma que a variável A passe a ter o valor da variável B e a variável B passe a ter o valor da variável A. Apresente os valores trocados.
- 3 Calcule o salário de um vendedor ao final do mês. O salário final será o salário fixo mais 15% sobre as vendas efetuadas. Os dados de entrada serão o salário fixo e o valor total das vendas efetuadas no mês.
- 4 Leia um valor que foi depositado na poupança e exiba o valor com o rendimento após um mês. Considere que a taxa de juros da poupança é de 0,7% ao mês.
- 5 Leia um valor e armazene em x, depois calcule e mostre o resultado da seguinte equação:

$$y = \frac{2 * \sqrt{x+1}}{3x^3}$$

- 6 Leia o valor pelo teclado e informe o valor da raiz quadrada do referido valor.
- 7 Leia o valor de um ângulo, converta de graus para radianos, calcule e mostre o valor do seno e do cosseno desse ângulo. Utilize a constante M\_PI, da biblioteca math.h, que fornece o valor de .  
Lembre-se:  
180 graus ---- pi radianos  
X graus ---- Y radianos
- 8 Leia o valor da base e do expoente, depois calcule e mostre o valor da potência.