



Aula 11

Prof: Henrique Augusto Maltauro

Desenvolvendo Algoritmos

C# (C Sharp)

- Comentários

Definidos por `//` para comentários de uma única linha.

Definidos por `/* */` para comentários de várias linhas.

C# (C Sharp)

- Comentários

```
// Isso é um comentário de uma única linha
```

```
/*
```

```
Isso é um comentário de várias linhas
```

```
*/
```

C# (C Sharp)

- Console

É uma classe do C# que representa os fluxos de entrada, saída e erro padrão para aplicativos de console.

C# (C Sharp)

- Console.WriteLine

Escreve no terminal da aplicação, alguma informação.

C# (C Sharp)

- Console.WriteLine

```
Console.WriteLine(12);  
// Saída: 12  
Console.WriteLine("Hello World");  
// Saída: Hello World
```

C# (C Sharp)

- `Console.ReadKey`

Lê o próximo caractere ou tecla de função pressionada pelo usuário.

C# (C Sharp)

- Console.ReadKey

```
Console.ReadKey();
```


C# (C Sharp)

- Console.ReadLine

Lê uma linha de caracteres digitadas no terminal da aplicação, e retorna uma string com a informação digitada.

C# (C Sharp)

- Console.ReadLine

```
string texto;  
texto = Console.ReadLine();
```

C# (C Sharp)

- Método `.Parse`

Converte uma string em outro valor definido antes do método.

Tipo da Conversão

```
int.Parse("123");
```



String a ser convertida

C# (C Sharp)

- Método `.Parse`

```
int valorInt = int.Parse("123");  
long valorLong = long.Parse("9787642");  
float valorFloat = float.Parse("15.7");  
double valorDouble = double.Parse("13.0");  
bool valorBool = bool.Parse("true");
```

Operadores

Operadores

O C# fornece vários operadores, muitos deles têm suporte aos tipos internos e permitem que você execute operações básicas com valores desses tipos.

Esses operadores incluem os seguintes grupos:

- Operadores Aritméticos.
- Operadores de Comparação.
- Operadores Lógicos Booleanos.
- Operadores de Igualdade.

Operadores

- C#: Operadores Aritméticos

São operadores que executam operações aritméticas com valores numéricos.

Operadores

- C#: Operadores Aritméticos

+ (adição)

- (subtração)

* (multiplicação)

/ (divisão)

Operadores

- C#: Operadores Aritméticos (adição)

```
Console.WriteLine(5 + 4);           // Saída: 9  
Console.WriteLine(5 + 4.3);         // Saída: 9.3  
Console.WriteLine(5.1m + 4.2m);     // Saída: 9.3
```

Operadores

- C#: Operadores Aritméticos (subtração)

```
Console.WriteLine(47 - 3);           // Saída: 44  
Console.WriteLine(5 - 4.3);          // Saída: 0.7  
Console.WriteLine(7.5m - 2.3m);      // Saída: 5.2
```

Operadores

- C#: Operadores Aritméticos (multiplicação)

```
Console.WriteLine(5 * 2);           // Saída: 10  
Console.WriteLine(0.5 * 2.5);       // Saída: 1.25  
Console.WriteLine(0.1m * 23.4m);    // Saída: 2.34
```

Operadores

- C#: Operadores Aritméticos (divisão)

```
Console.WriteLine(13 / 5);    // Saída: 2  
Console.WriteLine(13 / 5.0); // Saída: 2.6
```

Operadores

- C#: Operadores Aritméticos (divisão)

```
Console.WriteLine(16.8f / 4.1f);  
// Saída: 4.097561  
Console.WriteLine(16.8d / 4.1d);  
// Saída: 4.09756097560976  
Console.WriteLine(16.8f / 4.1f);  
// Saída: 4.097560975609756097560975609756098
```

Operadores

- C#: Operadores Aritméticos

% (módulo (Resto da divisão))

++ (incremento)

-- (decremento)

Operadores

- C#: Operadores Aritméticos (módulo)

```
Console.WriteLine(5 % 4);    // Saída: 1
Console.WriteLine(5 % -4);   // Saída: 1
Console.WriteLine(-5 % 4);   // Saída: -1
Console.WriteLine(-5 % -4);  // Saída: -1
```

Operadores

- C#: Operadores Aritméticos (módulo)

```
Console.WriteLine(-5.2f % 2.0f); // Saída: -1.2  
Console.WriteLine(5.9 % 3.1);    // Saída: 2.8  
Console.WriteLine(5.9m % 3.1m);  // Saída: 2.8
```


Operadores

- C#: Operadores Aritméticos (incremento)

```
int i = 3;  
Console.WriteLine(i);    // Saída: 3  
Console.WriteLine(i++);  // Saída: 3  
Console.WriteLine(i);    // Saída: 4
```

Operadores

- C#: Operadores Aritméticos (incremento)

```
double a = 1.5;  
Console.WriteLine(a);    // Saída: 1.5  
Console.WriteLine(++a);  // Saída: 2.5  
Console.WriteLine(a);    // Saída: 2.5
```

Operadores

- C#: Operadores Aritméticos (decremento)

```
int i = 3;  
Console.WriteLine(i);    // Saída: 3  
Console.WriteLine(i--);  // Saída: 3  
Console.WriteLine(i);    // Saída: 2
```

Operadores

- C#: Operadores Aritméticos (decremento)

```
double a = 1.5;  
Console.WriteLine(a);    // Saída: 1.5  
Console.WriteLine(--a);  // Saída: 0.5  
Console.WriteLine(a);    // Saída: 0.5
```

Operadores

- C#: Operadores de Comparação

São operadores que comparam valores numéricos.

Operadores

- C#: Operadores de Comparação

< (menor que)

> (maior que)

<= (menor ou igual)

>= (maior ou igual)

Operadores

- C#: Operadores de Comparação (menor que)

```
Console.WriteLine(7.0 < 5.1); // Saída: False  
Console.WriteLine(5.1 < 5.1); // Saída: False  
Console.WriteLine(0.0 < 5.1); // Saída: True
```

Operadores

- C#: Operadores de Comparação (maior que)

```
Console.WriteLine(7.0 > 5.1); // Saída: True  
Console.WriteLine(5.1 > 5.1); // Saída: False  
Console.WriteLine(0.0 > 5.1); // Saída: False
```


Operadores

- C#: Operadores de Comparação (menor ou igual)

```
Console.WriteLine(7.0 <= 5.1); // Saída: False  
Console.WriteLine(5.1 <= 5.1); // Saída: True  
Console.WriteLine(0.0 <= 5.1); // Saída: True
```

Operadores

- C#: Operadores de Comparação (maior ou igual)

```
Console.WriteLine(7.0 >= 5.1); // Saída: True  
Console.WriteLine(5.1 >= 5.1); // Saída: True  
Console.WriteLine(0.0 >= 5.1); // Saída: False
```

Operadores

- C#: Operadores Lógicos Booleanos

São operadores que executam valores lógicos com valores `bool`.

Operadores

- C#: Operadores Lógicos Booleanos

! (negação lógica)

& (AND lógico)

| (OR lógico)

^ (XOR lógico)

Operadores

- C#: Operadores Lógicos Booleanos (negação lógica)

```
bool passed = false;  
Console.WriteLine(!passed); // Saída: True  
Console.WriteLine(!true);   // Saída: False
```

Operadores

- C#: Operadores Lógicos Booleanos (AND lógico)

```
Console.WriteLine(true & true);    // Saída: True
Console.WriteLine(false & true);   // Saída: False
Console.WriteLine(true & false);   // Saída: False
Console.WriteLine(false & false);  // Saída: False
```

Operadores

- C#: Operadores Lógicos Booleanos (OR lógico)

```
Console.WriteLine(true | true);    // Saída: True
Console.WriteLine(false | true);   // Saída: True
Console.WriteLine(true | false);   // Saída: True
Console.WriteLine(false | false);  // Saída: False
```

Operadores

- C#: Operadores Lógicos Booleanos (XOR lógico)

```
Console.WriteLine(true ^ true);    // Saída: False  
Console.WriteLine(false ^ true);   // Saída: True  
Console.WriteLine(true ^ false);   // Saída: True  
Console.WriteLine(false ^ false);  // Saída: False
```


Operadores

- C#: Operadores Lógicos Booleanos

&& (AND lógico condicional)

|| (OR lógico condicional)

Operadores

- C#: Operadores Lógicos Booleanos (AND lógico condicional)

```
Console.WriteLine(true && true);    // Saída: True
Console.WriteLine(false && true);   // Saída: False
Console.WriteLine(true && false);   // Saída: False
Console.WriteLine(false && false);  // Saída: False
```

Operadores

- C#: Operadores Lógicos Booleanos (OR lógico condicional)

```
Console.WriteLine(true || true);    // Saída: True
Console.WriteLine(false || true);   // Saída: True
Console.WriteLine(true || false);   // Saída: True
Console.WriteLine(false || false);  // Saída: False
```

Operadores

- C#: Operadores de Igualdade

São operadores que verificam se os valores são iguais ou não.

Operadores

- C#: Operadores de Igualdade

`==` (igualdade)

`!=` (desigualdade)

Operadores

- C#: Operadores de Igualdade (igualdade)

```
int a = 1 + 2 + 3;  
Console.WriteLine(a == 6);      // Saída: True  
Console.WriteLine('a' == "A"); // Saída: False
```

Operadores

- C#: Operadores de Igualdade (desigualdade)

```
int a = 1 + 2 + 3 + 4;  
Console.WriteLine(a != 6);           // Saída: True  
Console.WriteLine("Hello" != "Hello"); // Saída: False
```

Exercício

Exercício

1 - Receber dois valores **double** e apresentar no console a adição, subtração, divisão e multiplicação dos dois valores.