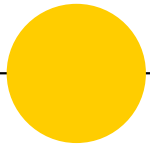


Introdução à Plataforma .NET



Profa. Luciana Balieiro Cosme

Ciência da Computação/IFNMG Montes Claros

Template SlidesCarnival

Operadores (mais comuns) e Precedência

Unário: + - ! ~ ++X --X

x=5; y = x++; ?

aritméticos: * / %

aritméticos: + -

relacional: < > <= >=

relacional: == !=

condicional: &&

condicional: ||

atribuição: = *= /= %= += -=

Associatividade

⦿ `int a = 13 / 5 / 2;`

● Funções lambda

⦿ operador =>

(parâmetros) => expressão

```
using System.Linq;
```

```
var numbers = new List<int>(new[] { 1, 2, 3 });
```

```
var b = numbers.Max(n => n * n);
```

```
Console.WriteLine(b);
```

Funções lambda

⦿ operador =>

```
Func<int, int> square = x => x * x;
```

```
Console.WriteLine(square(5));
```

ou

```
Console.WriteLine((Func<int, int>)  
(x => x * x)(5));
```

```
Func<int, int, int> adicao = (x, y) => (x + y);
```

```
Console.WriteLine(adicao(5, 1));
```

Exercícios

Escreva um programa em C# que receba dois números inteiros do usuário e exiba o resultado das seguintes operações: adição, subtração, multiplicação, divisão inteira e resto da divisão entre os números.



Exercícios

Escreva um programa em C# que utilize uma única expressão lambda para fazer todas as operações do exercício anterior. Em seguida, exiba o resultado na tela.



Referências

- Operadores => <https://learn.microsoft.com/pt-pt/dotnet/csharp/language-reference/operators/>
- Expressões lambda => <https://learn.microsoft.com/pt-pt/dotnet/csharp/language-reference/operators/lambda-expressions>
- Operador de coalescência nula => https://learn.microsoft.com/pt-br/dotnet/csharp/language-reference/operators/null-coalescing-operator?WT.mc_id=visualstudio-twitch-jefritz
- Calculadora => <https://learn.microsoft.com/pt-br/visualstudio/get-started/csharp/tutorial-console?view=vs-2022>