

# BÓSON TREINAMENTOS EM CIÊNCIA E TECNOLOGIA

Aprenda Ciências, Bancos de Dados, Linux, SQL, Excel, Lógica de Programação, Hardware, Eletrônica, Química, Arduino e muito mais!

[HOME](#) | [BANCOS DE DADOS E EXCEL](#) | [ELETRÔNICA](#) | [LINUX / MAC](#) | [PROGRAMAÇÃO](#) | [QUIZZES](#) | [REDES / HARDWARE](#) | [CARREIRA](#) | [CIÊNCIAS](#)

Início » Programação com C# » Números Complexos em C# - Introdução

## Números Complexos em C# – Introdução

Postado Em 10/02/2020 [data do post] por Fábio dos Reis [autor do post] em Programação com C# [categoria do post]

Curtir 0

Tweetar

### Números Complexos em C#

Neste artigo vamos mostrar como criar e efetuar **operações aritméticas básicas** com **Números Complexos** em C#.

### O Namespace System.Numerics

O **namespace System.Numerics** contém diversos tipos numéricos usados para complementar os tipos primitivos, como Int32 e Double, definidos no framework .NET.

Dentre as estruturas implementadas pelo namespace System.Numerics podemos citar: **Plane** (plano no espaço tridimensional), **Vector2** (vetor com dois valores), **Matrix4x4** (Matriz 4x4) e **Complex** (números complexos – objeto deste artigo), entre outras.

### A Estrutura Complex

A estrutura Complex, do namespace System.Numerics, permite representar números complexos e realizar diversas operações com esses números. Vamos começar mostrando como criar uma instância de um número complexo em C#.

### Como criar uma instância de um número complexo

Para criar uma instância de um número complexo em C# devemos passar dois valores do tipo *Double* para o construtor. O primeiro valor representa a parte real do número complexo, e o segundo valor representa a parte imaginária.

**Exemplo:** Vamos criar o número complexo **2+5i** em C#:

```
Complex c1 = new Complex(2,5);  
// Mostrar o valor do número complexo (forma padrão C#):  
Console.WriteLine(c1);
```

(2, 5)

### Pesquisar no Site

Para pesquisar, digite o que deseja e tecle Enter

### Assine nossa Newsletter

Email address:

Your email address

Sign up

### Curso de MySQL com Certificado e Material para Download



### Posts mais recentes

Erro O nome do tipo "SqlConnection" não pode ser encontrado no namespace "System.Data.SqlClient" (Resolvido) – Visual Studio  
10 comandos essenciais em SQL com exemplos de uso  
O que é um Dataset  
7 erros que todo programador deve evitar ao desenvolver software  
Modelagem de Dados – O que são Dependências em Bancos de Dados  
Type Annotations em Python (Type Hints)  
Exercício: Gerar Jogos da Mega Sena – Lógica de Programação  
Como usar Arrays em colunas no PostgreSQL

## Propriedades de Complex

Podemos exibir as partes real e imaginária de um número complexo com as respectivas propriedades *real* e *imaginary* de um objeto `Complex`:

```
Console.WriteLine("Parte real: " + c1.Real) ;
Console.WriteLine("Parte imaginária: " + c1.Imaginary);
```

Obs.: As propriedades `Real` e `Imaginary` são somente-leitura, de modo que não é possível modificar os valores de um objeto `Complex` já criado. Os métodos que executam operações em objetos `Complex` retornam um novo número complexo, caso o tipo de retorno seja `Complex` também.

É possível exibir o número complexo no formato matemático  $a + bi$  aplicando formatação com o método `Format` da classe `String`, como segue:

```
// Exibir o número no formato a+bi:
Console.WriteLine(String.Format("{0} + {1}i", c1.Real, c1.Imaginary));
```

## Operações com números complexos em C#

A estrutura `Complex` possui operadores que permitem realizar diversas operações aritméticas com números complexos, tais como adição, subtração, multiplicação, divisão e negação. Além disso, inclui métodos para executar outras operações numéricas, como exponenciação, raiz quadrada e valor absoluto de um número complexo, além de operações trigonométricas.

**Exemplo:** Vamos criar e somar dois números complexos. Para isso usaremos a operação `Add(num1,num2)`, onde:

- num1 = primeiro número complexo a adicionar
- num2 = segundo número complexo a adicionar

A adição de números complexos é realizada adicionando-se cada parte separadamente, de acordo com a fórmula:

$$(a+bi) + (c+di) = (a+c) + (b+d)i$$

Esta operação em C# retorna um valor do tipo `Complex`. Vejamos um exemplo:

```
Complex c1 = new Complex(2,5);
Complex c2 = new Complex(3, 11);
Complex c3 = Complex.Add(c1, c2);
Console.WriteLine("Soma: " + String.Format("{0} + {1}i", c3.Real,
c3.Imaginary));
```

Soma: 5 + 16i

Declaração MySQL REPLACE – Inserir ou atualizar linhas em uma tabela  
O que é um Faucet em criptomoedas  
Como desligar e reiniciar o Linux Debian 10 (Buster) e 11 (Bullseye)  
Comandos para sair de um script Python: `exit()`, `quit()`, `sys.exit()`, `os_exit()`  
Como testar uma fonte ATX com multímetro  
Pinagem do Conector ATX de 24 Pinos (ATX 2.2)  
Erro no Firefox “A execução do código não pode continuar porque VCRUNTIME140\_1.dll não foi encontrado” – RESOLVIDO  
Exercícios sobre Lei de Ohm – Física e Curso de Eletrônica  
Tipo enum em Java e classe de teste  
Vale a pena aprender linguagem C atualmente (2012 / 2022)?  
Dicionário de Bancos de Dados e Big Data  
Como saber há quanto tempo um computador com Windows 10 está ligado?  
Sets – Conjuntos em Python  
Como arredondar números em Python  
Como trabalhar com Transações em MySQL  
Como inserir dados em duas tabelas ao mesmo tempo no MySQL  
O que é Engenharia Social e como se prevenir de ataques  
O que é Biometria – Conceitos e Tecnologias  
Arredondar valores para cima no MySQL com função `CEILING()`  
Biblioteca PyMySQL para conexão do Python ao MySQL ou MariaDB  
Arredondar valores para baixo com a função `FLOOR()` no MySQL  
Como instalar o pacote AdoptOpenJDK no Linux Debian

## Tags

Android    Análise de Dados    Apple

Arduino

Bancos de Dados    C#

Carreira    CCNA    Certificação

Ciência de Dados    Criptografia

Elettricidade    Eletrônica

Eletrônica Digital    Ferramentas

Física    Hardware    IoT    Java

Linguagem C    Linguagem R    Linux

LPIC    Lógica

Lógica de Programação    Mac OS X

Maker    Matemática    MySQL

Notícias    Oracle    PHP

**Exemplo 02:** Multiplicação de números complexos em C#. Cada elemento do primeiro número é multiplicado por cada elemento do segundo número, segundo a fórmula:

$$(a+bi)(c+di) = ac + adi + bci + bdi^2$$

Vamos multiplicar os números  $2 + 5i$  e  $3 + 11i$  em C#. Para isso usaremos o método **Multiply(num1,num2)**, que recebe como argumentos dois números complexos e retorna como resultado o produto entre eles:

```
Complex c1 = new Complex(2,5);
Complex c2 = new Complex(3, 11);
Complex c3 = Complex.Multiply(c1, c2);
Console.WriteLine("Produto: " + String.Format("{0} + {1}i", c3.Real,
c3.Imaginary));
```

Produto: -49 + 37i

**Exemplo 03:** Agora vamos realizar uma subtração de números complexos. Para isso usaremos a operação **Subtract(num1,num2)**, onde:

- num1 = primeiro número complexo a adicionar
- num2 = segundo número complexo a adicionar

A subtração de números complexos é realizada subtraindo-se cada parte separadamente, de acordo com a fórmula:

$$(a+bi) - (c+di) = (a-c) + (b-d)i$$

Vamos ao um exemplo. Iremos subtrair o valor  $1 + 3i$  de  $3 + 5i$ :

```
Complex c1 = new Complex(1,3);
Complex c2 = new Complex(3, 5);
Complex c3 = Complex.Subtract(c2, c1);
Console.WriteLine("Subtração: " + String.Format("{0} + {1}i", c3.Real,
c3.Imaginary));
```

Subtração: 2 + 2i

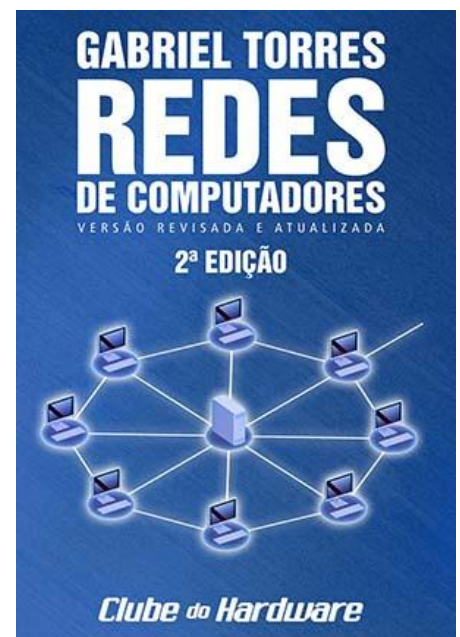
**Exemplo 04:** Vamos determinar o conjugado de um número complexo. Para isso, vamos empregar o método **Conjugate(complex)**, que recebe como argumento um número complexo e

## Vídeos sobre Carreira e Certificação

É possível aprender Tecno...



## Aprenda Redes de Computadores com o livro de Gabriel Torres:



retorna outro complexo como resultado.

O conjugado de um complexo é o número com o sinal trocado em sua parte imaginária.

Vamos determinar o conjugado de  $2 + 7i$  em C#:

```
Complex c1 = new Complex(1,3);  
Complex conj = Complex.Conjugate(c1);  
Console.WriteLine("Conjugado de {0}: {1}", String.Format("{0} + {1}i",  
c1.Real, c1.Imaginary), String.Format("{0} {1}i", conj.Real,  
conj.Imaginary));
```

```
Conjugado de 1 + 3i: 1 -3i
```

O conjugado de um complexo é empregado, por exemplo, para auxiliar na operação de divisão entre números complexos, que será nosso próximo exemplo.

**Exemplo 05:** Finalmente, vamos efetuar uma divisão entre dois complexos em C#. Usaremos o método *Divide(num1,num2)* para realizar tal operação, o qual divide o complexo num1 pelo complexo num2.

Vamos realizar a seguinte divisão de complexos com C#:

$$\frac{5 + 2i}{7 + 4i}$$

Código:

```
Complex c1 = new Complex(5, 2);  
Complex c2 = new Complex(7, 4);  
Complex cDiv = Complex.Divide(c1,c2);  
Console.WriteLine("Resultado da divisão: {0}", String.Format("{0} + {1}i",  
cDiv.Real, cDiv.Imaginary));
```

```
Resultado da divisão: 0,6615384615384615 + -0,09230769230769227i
```

É isso aí! A estrutura Complex em C# possui inúmeros outros métodos que implementam operações matemáticas com números complexos, se sua documentação oficial por ser consultada [aqui](#).

No próximo artigo vamos abordar algumas operações mais avançadas que podem ser realizadas com a estrutura Complex. Até!

C# Matemática Programação

Curtir 0

Tweetar



### Sobre Fábio dos Reis (1376 Artigos)

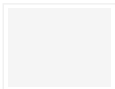
Fábio dos Reis trabalha com tecnologias variadas há mais de 25 anos, tendo atuado nos campos de Eletrônica, Telecomunicações, Programação de Computadores e Redes de Dados. É um entusiasta de Unix, Linux e Open Source em geral, adora Eletrônica e Música, e estuda idiomas, além de ministrar cursos e palestras sobre diversas tecnologias em São Paulo e outras cidades do Brasil.

Contato: | Website

← Artigo anterior

Próximo artigo →

## Artigos Relacionados



### Programação em C# - Arrays - Declarando e Instanciando - 16



### Leitura e gravação em arquivos com Python - Lendo arquivos de texto



### Curso de PHP - Funções para manipulação de strings - parte 01



### 05 - Lógica de Programação - Declaração e Atribuição de valores a variáveis



### Entrada de Dados em Python com função input()

## Escreva um comentário

Seu e-mail não será divulgado

Comentário

Nome\*

Email\*

Website

Publicar comentário

### Siga-nos nas Redes Sociais!



### Arquivos

Arquivos Selecionar o mês ▼

### Artigos Recentes

Erro O nome do tipo  
"SqlConnection" não pode ser  
encontrado no namespace  
"System.Data.SqlClient" (Resolvido) –  
Visual Studio  
10 comandos essenciais em SQL  
com exemplos de uso  
O que é um Dataset  
7 erros que todo programador deve  
evitar ao desenvolver software  
Modelagem de Dados – O que são  
Dependências em Bancos de Dados  
Type Annotations em Python (Type  
Hints)

### Buscar no Site

Para pesquisar, digite o que deseja e te