

Visual Studio 2019

Console C# (C Sharp)

Funções Matemática

O C# oferece inúmeras funções matemáticas, que vão ser útil em diversas áreas de atuação, desta forma poderemos desenvolver várias “rotinas” específicas de acordo com a aplicação solicitada.

Exemplos mais básico para nossa utilização:

Exemplo	Significado
<code>A = Math.Sqrt(x);</code>	Variável A recebe a raiz quadrada de x
<code>A = Math.Pow(x, y);</code>	Variável A recebe o resultado de x elevado a y
<code>A = Math.Abs(x);</code>	Variável A recebe o valor absoluto de x

- Lembrando que valor absoluto é chamado de modulo, ou seja, o valor sem sinal.

Exemplos para teste:

```
using System;
namespace EX_MATH
{
    class Program
    {
        static void Main(string[] args)
        {
            //Conteúdo
        }
    }
}
```

Conteúdo do “static void Main(string[] args)”

```
//declaramos umas variáveis
```

```
double a = 9.0;
```

```
double b = 2.0;
```

```
double x = 0.0;
```

```
Console.WriteLine("Raiz quadrada de " + a + " é " + Math.Sqrt(a));
```

```
Console.WriteLine("O numero " + a + " elevado à " + b + " é " + Math.Pow(a, b));
```

```
//Calculo
```

```
x = -1.0 * a;
```

```
Console.WriteLine("O valor absoluto de " + x + " é " + Math.Abs(x));
```

```
Console.ReadLine();
```

Vamos elaborar um solução:

Agora vamos elaborar um projeto que resolva uma equação de segundo grau, usando a fórmula de Bhaskara (método resolutivo para equações do segundo grau cujo nome homenageia o grande matemático indiano). Essa fórmula nada mais é do que um método para encontrar as raízes “reais” (não podemos extrair a raiz quadrada de números negativos) de uma equação do segundo grau fazendo uso apenas de seus coeficientes. Vale lembrar que coeficiente é o número que multiplica uma incógnita em uma equação.

- Em matemática, uma incógnita é uma variável cujo valor deve ser determinado de forma a resolver uma equação ou inequação. Normalmente, é representada pelas letras x , y e z , e as constantes pelas primeiras letras do alfabeto. Ex: $3x + 4 = 19$; $x - y = 6$.

Formula:

$$x = \frac{-b \pm \sqrt{\Delta}}{2.a}$$

$$\Delta = b^2 - 4ac$$

Para isso vamos declarar as variáveis para nosso programa:

- double a, b, c, x1, x2, delta;

Vamos resolver a 1ª operação: $x^2 + 5x - 2 = 0$, portanto teremos:

$a=1$, $b=5$ e $c=-2$, qual vai ser o resultado de $x'(x1)$ e $x''(x2)$?

Codificação:

```
using System;
```

```
namespace EX_BHASKARA
```

```
{
```

```
    class Program
```

```
    {
```

```
        static void Main(string[] args)
```

```
        {
```

```
            //Conteúdo
```

```
        }
```

```
    }
```

```
}
```

Conteúdo do Codificação (Equação de 2º Grau)

```
double a, b, c, x1, x2, delta;
Console.WriteLine("BHASKARA");
a = double.Parse(Console.ReadLine());
b = double.Parse(Console.ReadLine());
c = double.Parse(Console.ReadLine());
//Calcular delta
delta = Math.Pow(b, 2.0) - (4 * a * c); // Podemos calcular:  $\Delta = (b^2) - (4 \cdot a \cdot c)$ 
// Este valor não pode ser negativo
// Futuramente podemos tratar (Consistências)
//Calculando x1 e x2
x1 = (-b + Math.Sqrt(delta)) / (2.0 * a);
x2 = (-b - Math.Sqrt(delta)) / (2.0 * a);
//Exibindo os Resultados...
Console.WriteLine("Delta = " + delta);
Console.WriteLine("X'  = " + x1);
Console.WriteLine("X'' = " + x2);

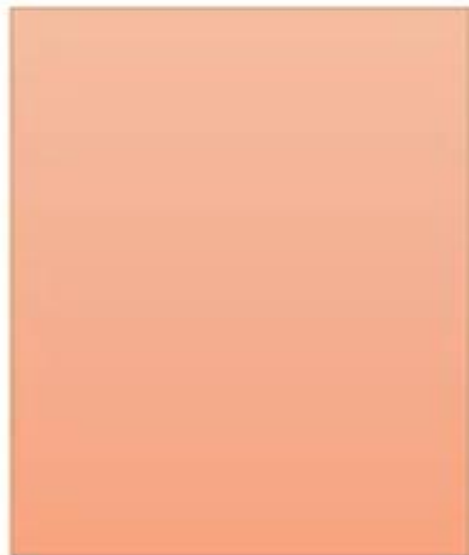
Console.ReadLine(); //Este ReadLine é apenas para aguardar o usuário pressionar “enter”, após o programa é encerrado.
```


Desenvolva:

Fazer um programa para ler as medidas da largura e comprimento de um terreno retangular com uma casa decimal, bem como o valor do metro quadrado do terreno com duas casas decimais. Em seguida, o programa deve mostrar o valor da área do terreno, bem como o valor do preço do terreno, ambos com duas casas decimais, conforme exemplo.

Exemplo:

Entrada:	Saída:
10.0 30.0 200.00	AREA = 300.00 PRECO = 60000.00



10.0 metros

30.0 metros

$$\text{área} = \text{largura} \times \text{comprimento}$$

$$\text{preço} = \text{área} \times \text{preço do metro quadrado}$$

Codificação:

```
using System;
```

```
namespace EX_TERRENO
```

```
{
```

```
    class Program
```

```
    {
```

```
        static void Main(string[] args)
```

```
        {
```

```
            // Conteúdo
```

```
        }
```

```
    }
```

```
}
```

Conteúdo (EX_Terreno)

```
double largura, comprimento, precoM2, area, precoTerreno;
```

```
Console.Write("Largura: ");
```

```
largura = double.Parse(Console.ReadLine());
```

```
Console.Write("Comprimento: ");
```

```
comprimento = double.Parse(Console.ReadLine());
```

```
Console.Write("Preço por m2 : R$ ");
```

```
precoM2 = double.Parse(Console.ReadLine());
```

```
area = largura * comprimento;
```

```
precoTerreno = precoM2 * area;
```

```
Console.WriteLine("Area de : " + area.ToString("F2") + " m2");
```

```
Console.WriteLine("Preço do Terreno: R$ " + precoTerreno.ToString("F2"));
```

```
Console.ReadLine();
```

Quando for necessário trocar a “,” por “.” no formato de saída (números com casas decimais), não podemos esquecer:

Inserir :

```
using System.Globalization;
```

```
...
```

```
Console.WriteLine("Area de : " + area.ToString("F2", CultureInfo.InvariantCulture) + " m²");
```

```
Console.WriteLine("Preço do Terreno: R$ " + precoTerreno.ToString("F2", CultureInfo.InvariantCulture));
```

```
....
```

Debug o Código (IDE Visual Studio “C#”)

Como executar o debug do Visual Studio

- Para marcar uma linha de breakpoint:
 - Depurar -> Alternar ponto de interrupção (F9)
- Para iniciar o debug:
 - Depurar -> Iniciar depuração (F5)
- Para executar uma linha:
 - F10
- Para interromper o debug:
 - SHIFT + F5 