

ESCOLA DE TECNOLOGIA DA INFORMAÇÃO
TÓPICOS ESPECIAIS DE SISTEMAS

Programação Básica C# utilizando classes

Geucimar Briatore
geucimar@up.edu.br

Programação Básica C# utilizando classes

```
/******
```

1. Tipos de variáveis var, int, short, long, decimal, double, bool e string;
2. Operadores de atribuição, aritméticos, relacionais, lógicos e unários;
3. Estruturas de decisão if/else, ternárias e switch;
4. Estruturas de repetição while, do/while, for, foreach;
5. Arrays.

```
*****/
```

Comandos dotnet

```
dotnet new console
```

```
dotnet new console --use-program-main
```

```
dotnet build
```

```
dotnet clean
```

```
dotnet run
```

<https://learn.microsoft.com/pt-br/dotnet/core/tools/>

<https://learn.microsoft.com/pt-br/dotnet/core/tutorials/top-level-templates>

Olá mundo C#! (template .NET SDK 6+)

```
//See https://aka.ms/new-console-template for more information  
Console.WriteLine("Olá mundo C#!");
```

Olá mundo C#! (template .NET SDK 6-)

```
//Diretivas explícitas
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;

//Namespace
namespace Mundo {

    class Program {
        static void Main(string[] args) {
            Console.WriteLine("Olá mundo C#!");
        }
    }
}
```

Tipos de variáveis

Tipo / Exemplo	Bits	Descrição
<code>int</code> contador = 10;	32	Números inteiros
<code>long</code> contadorLongo = 10L;	64	Números inteiros (ext.)
<code>float</code> valor = 0.11F;	32	Números fracionários
<code>double</code> valor = 0.11;	64	Números fracionários (ext.)
<code>decimal</code> pagamento = 0.99M;	128	Valores monetários
<code>char</code> digito = 'A';	16	Caractere
<code>string</code> texto = "Texto longo";	-	Sequência de caracteres
<code>bool</code> isCorreto = false;	8	Booleano

Console.ReadLine() + Convert.To...()

// Converte string to int

```
int idade = Convert.ToInt32(Console.ReadLine());
```

// Converte string to double

```
double valor = Convert.ToDouble(Console.ReadLine());
```

Operadores básicos

- Operadores aritméticos: **+**, **-**, *****, **/** e **%**;
- Operadores unários: **++** e **--**;
- Operadores de atribuição: **=**, **+=**, **-=**, ***=**, **/=** e **%=**;
- Operadores relacionais: **==**, **!=**, **<**, **<=**, **>** e **>=**.

Operadores lógicos

Operador	Descrição
&&	E (and)
	OU (or)
!	NÃO (not)

```
if (isBomAluno && possuiFreq && possuiNota) {  
    //Faz alguma coisa  
} else if (isBomAluno && possuiNota && !possuiFreq  
    || isBomAluno && !possuiNota && possuiFreq) {  
    //Faz outra coisa  
}
```

Condicionais if/else e swicth

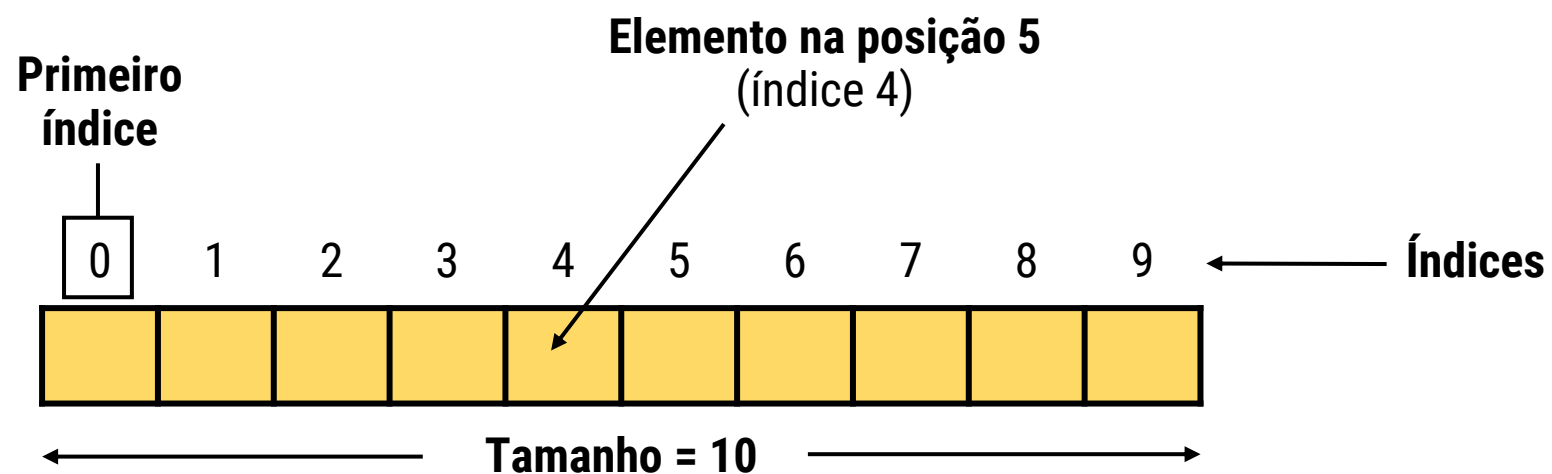
```
int nota = 50;

if (nota < 40)
{
    //Faz alguma coisa
}
else if (nota >= 70)
{
    //Faz outra coisa
}
```

```
int dia = 2;

switch (dia)
{
    case 1:
        //comando
        break;
    ...
    default:
        //comando
        break;
}
```

Vetores (arrays)



Vetor de números inteiros

//Criação

```
int[] vetor = new int[3];
```

//Atribuição

```
vetor[0] = 34;
```

```
vetor[1] = 27;
```

```
vetor[2] = 3;
```

//Leitura

```
int valor1 = vetor[0];
```

```
int valor2 = vetor[1];
```

```
int valor3 = vetor[2];
```

Iteradores for, while e do/while

```
for(int i = 0; i < 10; i++) {  
    //Faz alguma coisa  
}
```

```
int x= 1;  
while (x < 10) {  
    //Faz alguma coisa  
    x++;  
}
```

```
int x= 0;  
do {  
    //Faz pelo menos uma vez  
    x++;  
} while (x < 10)
```

Exercício de fixação 1

- Criar um programa para processar as seguintes informações de um aluno: matrícula, nome , nota1, nota2 e se ele está aprovado ou reprovado;
- Durante a execução calcular a nota final do aluno utilizando a seguinte regra: **notaFinal = (nota1 + nota2) / 2**. Se a nota final for igual ou superior a 6 o aluno está aprovado, caso contrário estará reprovado;
- O programa deverá imprimir o seguinte relatório:

Matrícula: **XXXXX**

Nome: **XXXXX XXXXX**

Aprovado: (**x**) Sim () Não

Nota final: **XXXXX**

Exercício de fixação 2

- Transformar o programa do **exercício 1** para um sistema que permita ler a entrada de dados pelo usuário em um vetor durante execução;
- Permitir que o usuário informe, primeiro, os dados de **5 (cinco) alunos** e depois de capturados os dados, imprimir o relatório final com todos os dados:

Matrícula: **xxxxx**

Nome: **xxxxx xxxxx**

Aprovado: (**x**) Sim () Não

Nota final: **xxxxx**

Exercício de fixação 3

- Criar um programa para processar as seguintes informações de um aluno: matrícula, nome , nota 1, nota 2, nota 3, nota 4 e nota 5;
- O sistema deverá ler as notas do aluno em um vetor e a entrada de dados pelo usuário só poderá ser nos valores de 0 a 10;
- No final da execução imprimir o seguinte relatório:

Matrícula: **xxxxxx**

Nome: **xxxxxx xxxxxx**

Nota (**x**) : **xxxxxx**

Nota (**y**) : **yyyyyy**

(...)