

**JOGOS DIGITAIS : PPI2**

## **a história da plataforma .NET**

Professor Sergio Souza Novak



# História do C# e .NET

- década de 1990 a Microsoft tinha diversas tecnologias.
- Problemas de Migração diante das plataformas

# História do C# e .NET

Solução inicial: Usar o **JAVA**

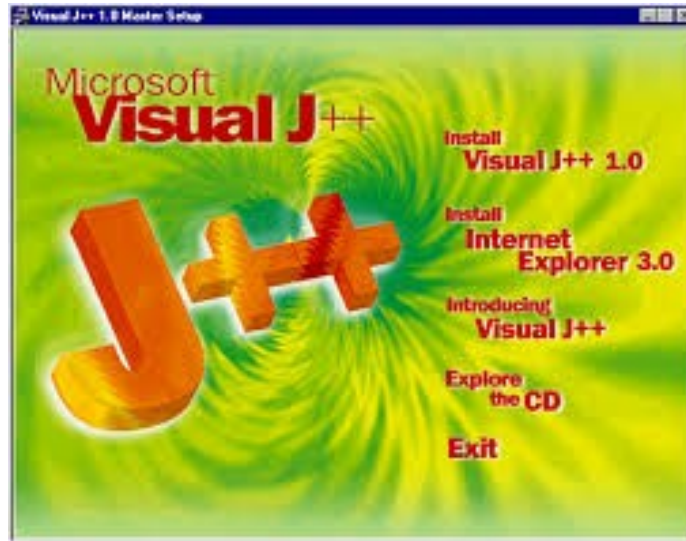


# História do C# e .NET

- JAVA não se comunicava bem com as bibliotecas de **código nativo** (código de máquina) que já existiam.
- Para resolver isso, a Microsoft decidiu criar a sua própria implementação do Java chamado **J++**

# História do C# e .NET

Solução básica: Estender o Java para **J++**





- O J++ era uma versão da linguagem Java que só podia ser executada no ambiente Microsoft. Seu código não podia ser executado em mais nenhum ambiente Java.
- Por isso a Microsoft foi processada.
- Uma das mais conhecidas batalhas judiciais da época.

# História do C# e .NET

Solução ótima: criar a plataforma .NET



# História do C# e .NET

- O .NET foi desde o início projetado para trabalhar com diversas linguagens de programação, assim diversas linguagens diferentes compartilhariam o mesmo conjunto de bibliotecas.
- Com isso, para um programador migrar de uma linguagem para outra ele precisaria apenas aprender a linguagem sem se preocupar com as bibliotecas e APIs.



# História do C# e .NET

- Em 2002, foi lançado a linguagem **C# 1.0**, junto com o ambiente .Net 1.0.
- **C#** (lê-se "cê xarpe") é uma linguagem de programação, **multiparadigma**, de **tipagem forte**, desenvolvida pela Microsoft como parte da plataforma .NET.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace ConsoleApp2
{
    class Program
    {
        static void Main(string[] args)
        {
            Point p = new Point(11, 11);
            Console.WriteLine(p.toString());
            soma1(p);
            Console.WriteLine(p.toString());

            Vector v = new Vector(13, 13);
            Console.WriteLine(v.toString());
            soma1(v);
            Console.WriteLine(v.toString());
        }
    }
}
```

# Exemplo de Código simples em C#

Disponível em

([github.com/SergioNoivak/PPI-2-curso-de-Jogos-Digitais/blob/master/passagem%20por%20copia%20e%20referencia%20dados%20compostos/Program.cs](https://github.com/SergioNoivak/PPI-2-curso-de-Jogos-Digitais/blob/master/passagem%20por%20copia%20e%20referencia%20dados%20compostos/Program.cs))

**JOGOS DIGITAIS : PPI2**

## **Objetivos GERAIS do C#(segundo ECMA)**

Professor Sergio Souza Novak



# Objetivos GERAIS do C#(segundo ECMA)

1. A linguagem C# destina-se a ser um **simples, moderna, de propósito geral, orientada a objetos.**



# Objetivos GERAIS do C#(segundo ECMA)

2. A linguagem C# deve ser **escalável** e **segura** propiciando a **produtividade** e a **durabilidade**.



# Objetivos GERAIS do C#(segundo ECMA)

3.A linguagem destina-se a ser utilizada no desenvolvimento de componentes de software adequados para implantação **em ambientes distribuídos.**



# Objetivos GERAIS do C#(segundo ECMA)

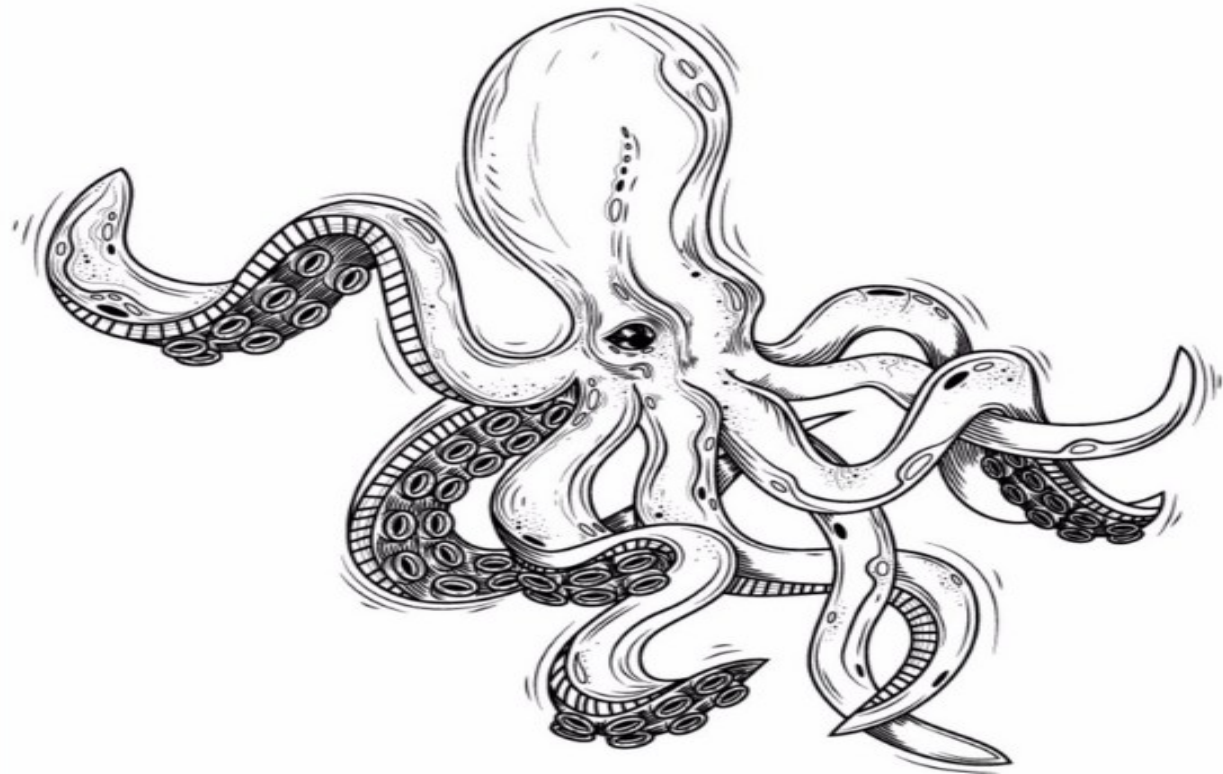
4. A **portabilidade** é muito importante para o código fonte, e programadores, especialmente aqueles já familiarizados com C e C++.



JOGOS DIGITAIS : PPI2

## A Máquina Virtual do C#: CLR

Professor Sergio Souza Novak





# A máquina Virtual do c# : CLR

- Em uma linguagem de programação como C e Pascal, temos a seguinte situação quando vamos compilar um programa:
- O código fonte é compilado para código de máquina **específico** de uma plataforma e sistema operacional.

# A máquina Virtual do c# : CLR

```
#define inicio_pinos 2
```

```
#define out_pinos 11
```

```
bool executar = true;
```

```
void setup() {
```

```
    for(int i=inicio_pinos ; i<out_pinos ;i++)
```

```
        pinMode(i,OUTPUT);
```

```
    Serial.begin(9600);
```

```
    pinMode(A0,INPUT);
```

```
}
```

Linguagem C  
embaracada  
para Arduíno

**Disponível em**  
([github.com/SergioNoivak/estudoArduinoUNO/blob/master/loops/loops.ino](https://github.com/SergioNoivak/estudoArduinoUNO/blob/master/loops/loops.ino))

# A máquina Virtual do c# : CLR

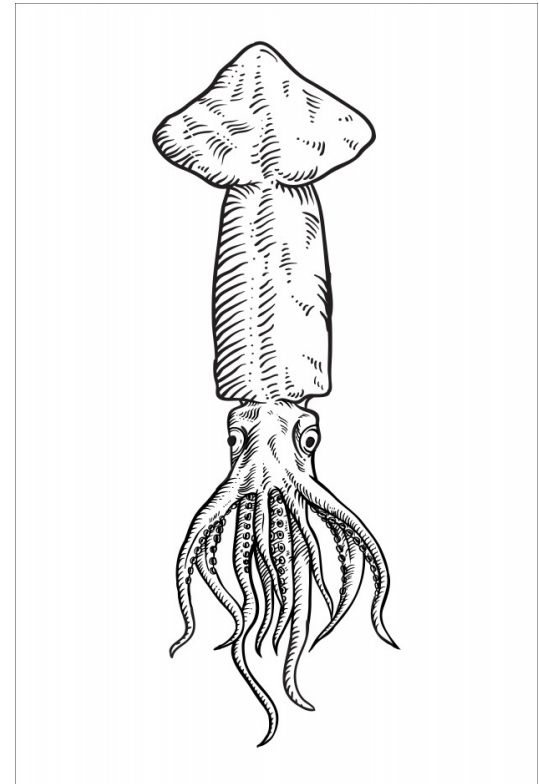
```
#include<stdio.h>
#include<string.h>
int main(){
    FILE* arquivo1 = fopen("novo_six.c", "r");
    FILE* arquivo2 = fopen("Untitled1.cpp", "r");
    FILE* arquivo_sintese = fopen("sintese.txt", "w");
    /*Caso do arquivo nulo*/
    if(arquivo1==NULL || arquivo2==NULL){
        printf("O arquivo nao pode ser encontrado");
        int c =getchar();
        fclose(arquivo1);
        fclose(arquivo2);
        fclose(arquivo_sintese);
        return 0;
    }
```

## Linguagem C para o Console Desktop

**Disponível em**  
([github.com/SergioNoivak/Algoritmos/blob/master/Arquivos\\_lista\\_02\\_03.c](https://github.com/SergioNoivak/Algoritmos/blob/master/Arquivos_lista_02_03.c))

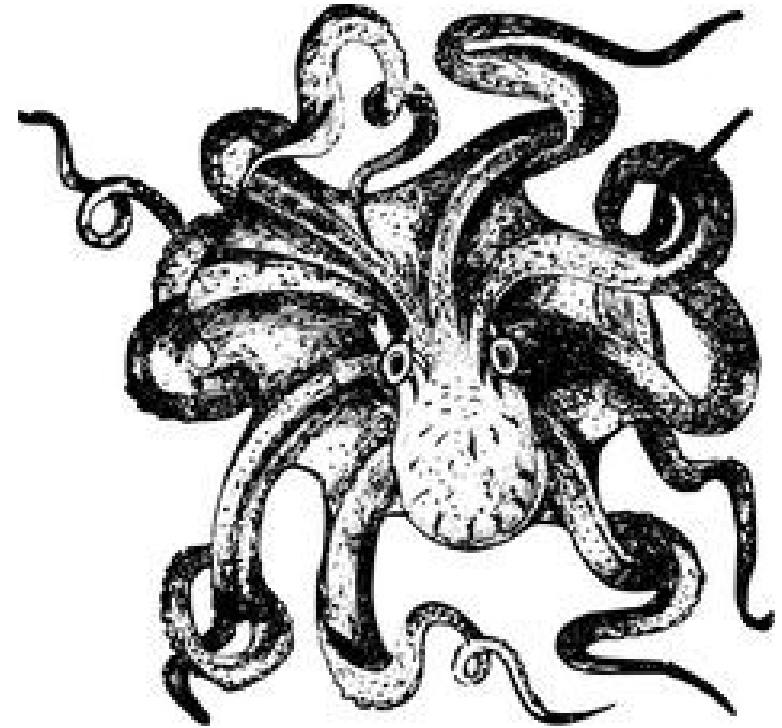
# A máquina Virtual do c# : CLR

- Muitas vezes o próprio código fonte é desenvolvido visando uma única plataforma!
- Precisamos reescrever um mesmo pedaço da aplicação para diferentes sistemas operacionais, já que eles não são compatíveis.



# A máquina Virtual do c# : CLR

- C# e a Máquina Virtual CLR
- CLR isola totalmente a aplicação do sistema operacional.
- A máquina virtual deve trabalhar com diversas linguagens de programação diferentes.



# A máquina Virtual do c# : CLR

**DESENHAR ESQUEMA .NET**

# A máquina Virtual do c# : CLR

- a CLR não pode executar diretamente o código do C#.
- ela precisa executar uma linguagem intermediária comum a todas as linguagens da plataforma .Net, a CIL(Common Intermediate Language).

# A máquina Virtual do c# : CLR

- Para gerar o CIL que será executado pela CLR, precisamos passar o código C# por um compilador da linguagem, como o programa `csc.exe`.



**JOGOS DIGITAIS : PPI2**

## **Seu Primeiro Programa em C#**

Professor Sergio Souza Novak



```
// A Hello World! program in C#.
```

```
using System;
```

```
namespace HelloWorld
```

```
{
```

```
    class Hello
```

```
    {
```

```
        static void Main()
```

```
        {
```

```
            Console.WriteLine("Hello World!");
```

```
            // Keep the console window open in debug mode.
```

```
            Console.WriteLine("Press any key to exit.");
```

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

```
        }
```

```
    }
```

```
}
```

O programa exibe  
uma string no  
console

Disponível em  
([github.com/SergioNoivak/PPI-2-  
curso-de-Jogos-Digitais/blob/ma-  
ster/hello\\_world.cs](https://github.com/SergioNoivak/PPI-2-curso-de-Jogos-Digitais/blob/master/hello_world.cs))

# Seu primeiro programa em C#

- Você pode invocar o compilador do C#, digitando o nome do seu arquivo executável (csc.exe) em um prompt de comando.

```
csc -out:My.exe File.cs
```

# Seu primeiro programa em C#

- Observações: verifique se o compilador do c# foi adicionado as variáveis de ambiente do sistema.
- Você pode escolher entre utilizar o Visual Studio e Compilar usando a interface gráfica ou usar o compilador via linha de comando no cmd do Windows.