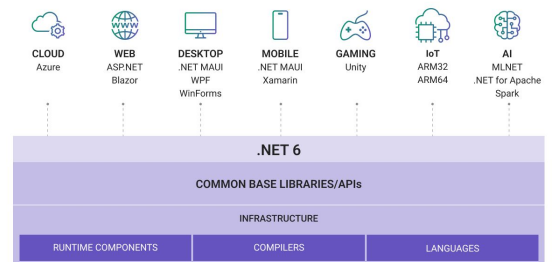


# Introdução à Programação

Introdução C#  
Entrada e saída  
Variáveis  
Operadores



## Unified Development Platform



Source: Microsoft, 2023

## Top .NET Programming Languages

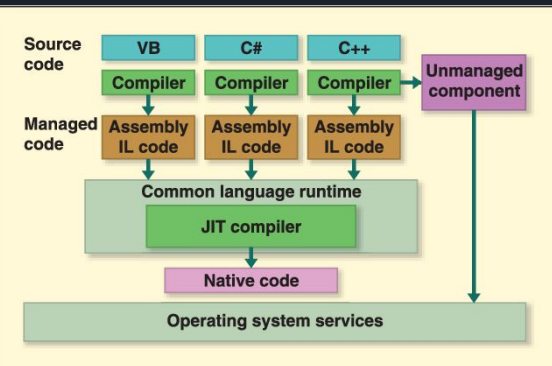


Fig. 2. CLR Execution Model.

# Instalações



C#

```
class Program
{
    0 references
    static void Main(string[] args)
    {
        System.Console.WriteLine();
    }
}
```

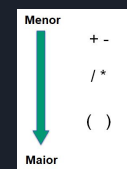
- > { } / local
- > ;
- > Tabulação
- > Case Sensitive
- > CamelCase
- > Nome da classe:  
maiuscula = nome  
do arquivo.
- > Executável: Main

# Prática



## Hello world

- Write/Line
- comentários
- erros no código
- Operadores aritméticos



## Hello world

- `Console.WriteLine("Hello");`
- `Console.WriteLine("Hello" + "World");`
- `Console.WriteLine("{0} + {1} = {2}", firstNumber, secondNumber, result);`  
`string word = "World";`
- `Console.WriteLine($"Hello {word}");`



## Operadores

Operator	Name	Description
+	Addition	Adds together two values
-	Subtraction	Subtracts one value from another
*	Multiplication	Multiplies two values
/	Division	Divides one value by another
%	Modulus	Returns the division remainder
++	Increment	Increases the value of a variable by 1
--	Decrement	Decreases the value of a variable by 1

## Exercício



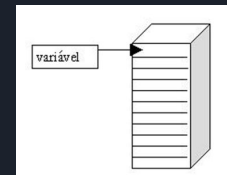
Escreva um programa para exibir o resultado das 4 operações matemáticas básicas (+, -, \*, /) entre os números 15 e 7.

## variáveis

Uma área de memória para guardar informação.

Alguns tipos (básicos) de dados:

int / double / bool / string



• Integer Types	
- byte:	■
- short:	■ ■
- int:	■ ■ ■ ■
- long:	■ ■ ■ ■ ■ ■ ■ ■
• Floating Point Types	
- float:	■ ■ ■ ■
- double:	■ ■ ■ ■ ■ ■ ■ ■
• Other Types	
- boolean:	■
- char:	■ ■



## Tipos de dados

Tipo	Tamanho	Valores Possíveis
bool	1 byte	true e false
byte	1 byte	0 a 255
sbyte	1 byte	-128 a 127
short	2 bytes	-32768 a 32767
ushort	2 bytes	0 a 65535
int	4 bytes	-2147483648 a 2147483647
uint	4 bytes	0 to 4294967295
long	8 bytes	-9223372036854775808L to 9223372036854775807L
ulong	8 bytes	0 a 18446744073709551615
float	4 bytes	Números até 10 elevado a 38. Exemplo: 10.0f, 12.5f
double	8 bytes	Números até 10 elevado a 308. Exemplo: 10.0, 12.33
decimal	16 bytes	números com até 28 casas decimais. Exemplo 10.991m, 33.333m
char	2 bytes	Caracteres delimitados por aspas simples. Exemplo: 'a', 'ç', 'o'

## Nomes de variáveis

- Deve iniciar com uma letra ou "\_"
- Após a primeira letra pode ter várias outras letras ou números, ou o símbolo "\$"
- Não pode conter espaço (" ") e nenhum outro símbolo especial, como "@", "&", "%", etc.
- Não pode ser igual a nenhuma das palavras chave da linguagem, ou seja, não pode ter o nome de uma instrução.
- Boa prática: use nomes significativos



## Type Casting

// implicit


```
int myInt = 9;
```

```
double myDouble = myInt;
```

// explicit

```
double myDouble = 9.78;
```

```
int myInt = Convert.ToInt32( myDouble);
```



## Entrada de datos

```
string userName = Console.ReadLine( );
```

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



## Exercício



Faça um programa que leia duas notas de um aluno, calcule e mostre a média aritmética do aluno.

## Exercício



Faça um programa que leia o salário de um funcionário. Sabendo que o salário do funcionário teve um aumento de 25%, calcular e mostrar o novo salário.