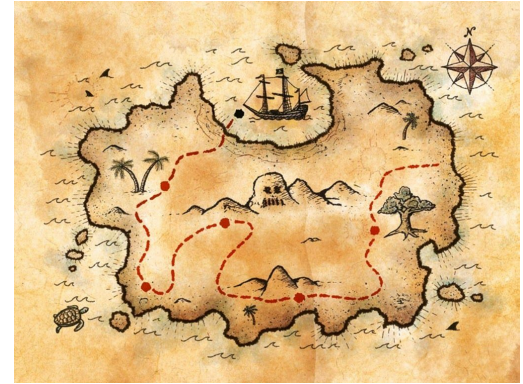
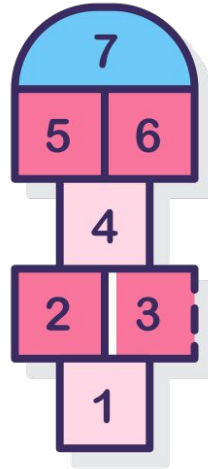
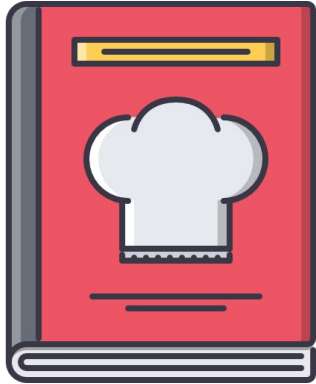


# Lógica de Programação

# Algoritmo

- Sequência **finita de passos** que levam à execução de uma tarefa
- Algo muito comum no nosso dia a dia, sendo de TI ou não =D



# Programa

- **Algoritmo** escrito em uma **linguagem** de programação

# Linguagem natural



## Linguagem de máquina

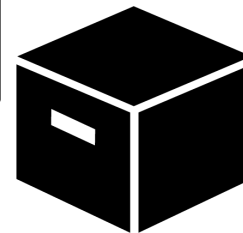


# Linguagem de programação



# Programador

# Compilador



# E qual linguagem usaremos neste curso?

- Neste nivelamento o foco é entender os princípios da programação. Assim, utilizamos a ferramenta Portugol Studio, que possui uma linguagem própria que aproxima a linguagem de programação ao português!

{Portugol  Studio}

# Estrutura inicial de um código em Portugol

programa

{

/\* Declaração de variáveis, estruturas e outras funções \*/

funcao inicio ()

{

/\*Execução da função início\*/

}

}

# Fluxograma - Outra forma de representação

- É uma forma universal de representação, pois se utiliza de figuras geométricas para ilustrar passos a serem seguidos para a resolução de problemas



Indica o início ou fim do algoritmo



Indica o sentido do fluxo de dados



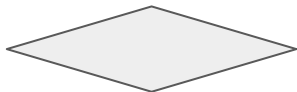
Indica cálculos e atribuições de valores



Representa a entrada de dados



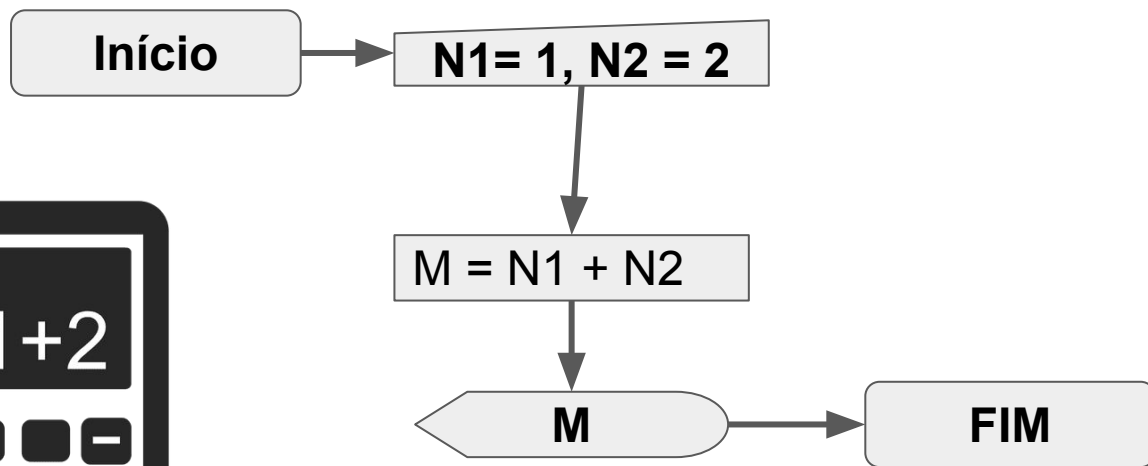
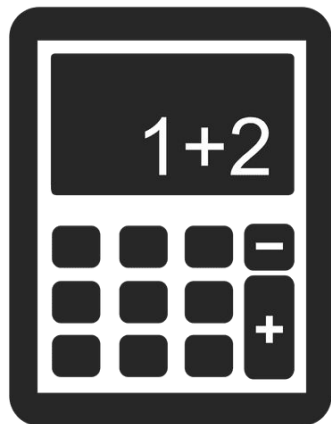
Representa a saída de dados



Tomada de decisão

# Fluxograma

Exemplo : Soma de 2 números



# O que iremos aprender :

- Operações de entrada e saída
- O que são variáveis e constantes
- Desvios condicionais ( se e senão )
- Operadores Lógicos ( E, OU ... )
- Laços de repetição ( enquanto, para ... até )
- Estruturas de dados ( Vetores, Matrizes, Filas e Pilhas )
- Subrotinas ( Funções )
  - Recursividade
  - Bibliotecas

**Com exemplos práticos!!**





Mas antes, vamos explorar o Portugal Studio

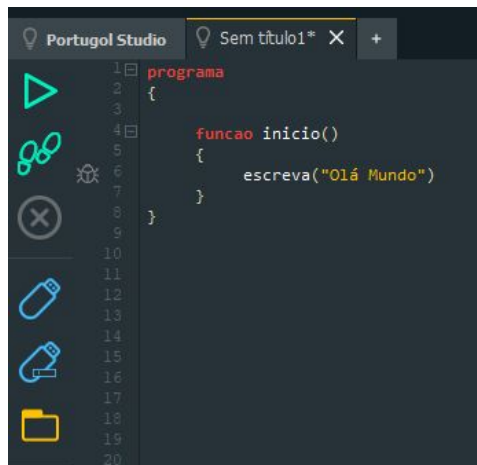


# Nosso primeiro programa: Olá mundo!

Execute no Portugol Studio o código : Olá Mundo

- O que esse código faz?
- Quais dificuldades vocês tiveram em entender este trecho de código?

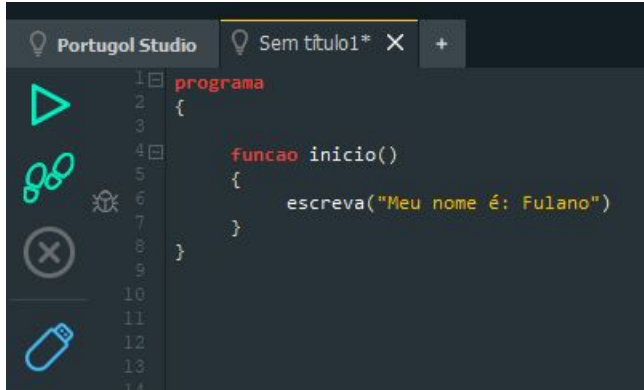
<HELLO WORLD />



```
1 programa
2 {
3
4     funcao inicio()
5     {
6         escreva("Olá Mundo")
7     }
8 }
9
10
11
12
13
14
15
16
17
18
19
20
```

# Identificando-se : Qual é o seu nome?

Execute os seguintes programas - **‘Meu nome é: Fulano’**



```
1 programa
2 {
3
4     funcao inicio()
5     {
6         escreva("Meu nome é: Fulano")
7     }
8
9
10
11
12
13
14
```



# Operações de entrada e saída

Fazendo a leitura do nome a partir do teclado.

programa

{

funcao inicio ()

{

cadeia nome /\*cadeia se refere ao tipo da variável que é uma cadeia de caracteres\*/

escreva("Digite seu nome: ")

leia(nome)

escreva("Seu nome é : ", "\n", nome )

}

}

Falaremos depois  
sobre **variáveis** e  
seus **tipos** !!

# Operações de entrada e saída

Quando escrevemos :

**cadeia** nome

**leia**(nome)

**leia** é uma operação de **entrada** que permite que o que escrevemos no teclado seja **lido e armazenado** na variável “**nome**”. Logo estamos **entrando** com uma informação no programada durante sua execução.

# Por que **entrada** e **saída**?

Quando escrevemos :

```
cadeia nome = Maria
```

```
escreva("Meu nome é: ", nome)
```

**escreva** é uma operação de **saída** que permite que a informação escrita entre seus parênteses “()” seja apresentado na tela do computador, logo como é uma informação de apresentação, entendemos como uma informação de **saída**.

# Voltando ao programa anterior...

Fazendo a leitura do nome a partir do teclado.

```
programa
```

```
{
```

```
    funcao inicio ()
```

```
{
```

```
    cadeia nome /*cadeia se refere ao tipo da variável que é uma cadeia de caracteres*/
```

```
    escreva("Digite seu nome: ")
```

```
    leia(nome)
```

```
    escreva("Seu nome é : ", "\n", nome )
```

```
}
```

```
}
```

Ficou faltando entendermos o que são as **variáveis**!



# Variáveis e constantes

- **Variáveis e constantes** representam uma posição na memória, onde pode ser armazenado um **único dado** ( valor ).
- Possuem **tipo**, **nome** e um **valor**
- A diferença entre variáveis e constantes é que enquanto o **valor da variável pode mudar** durante a execução do programa o **valor da constante não**.

```
{  
cadeia nome = "Raul"  
escreva (nome)  
nome = "Pedro"  
escreva ("\n",nome)  
}
```

O que será  
impresso na tela?





# Variáveis e constantes

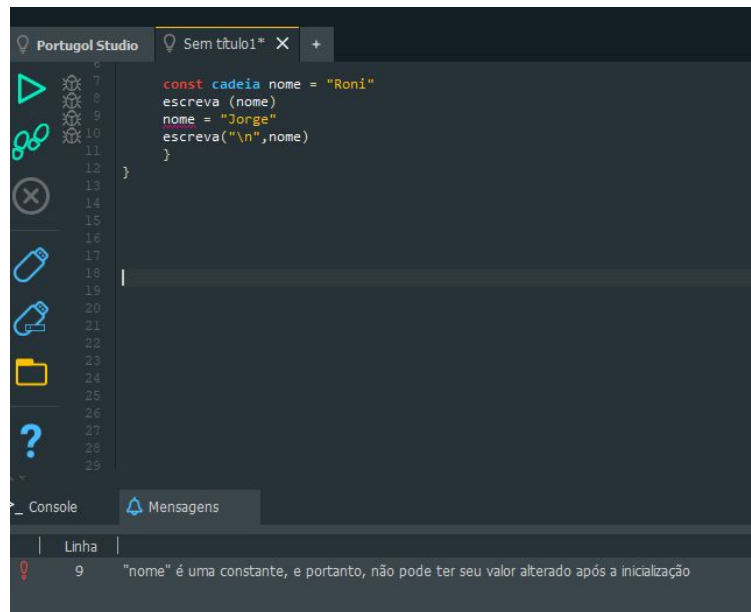
- Em algumas linguagens ( incluindo Portugol ) as variáveis podem ser tipadas, ou seja, aceitam apenas valores referentes ao seu tipo, representado antes do nome da variável:
  - Tipos de variáveis na linguagem do Portugol Studio
    - inteiro : Número inteiros -> 1 ; 2 ; 3
    - real : Números de ponto flutuante -> 1.1 ; 3.14 ; 10.3
    - cadeia : Cadeia de caracteres -> “Adoro estudar programação”
    - caracter : Apenas um caractere -> “A”, “1”
    - logico : Caractere booleano : verdadeiro, falso

# Variáveis e constantes

- Finalmente, para declarar uma **constante** basta colocar o indicador **const** antes da declaração da constante

**const** cadeia nome = Raul  
escreva (nome)  
nome = Pedro  
escreva (nome)

O que será impresso na tela?



The screenshot shows the Portugol Studio IDE. The code editor contains the following code:

```
const cadeia nome = "Roni"  
escreva (nome)  
nome = "Jorge"  
escreva ("\n", nome)  
}
```

The console output at the bottom shows the following message:

9 "nome" é uma constante, e portanto, não pode ter seu valor alterado após a inicialização

# Exemplo Variáveis

inteiro idade

real peso

real altura

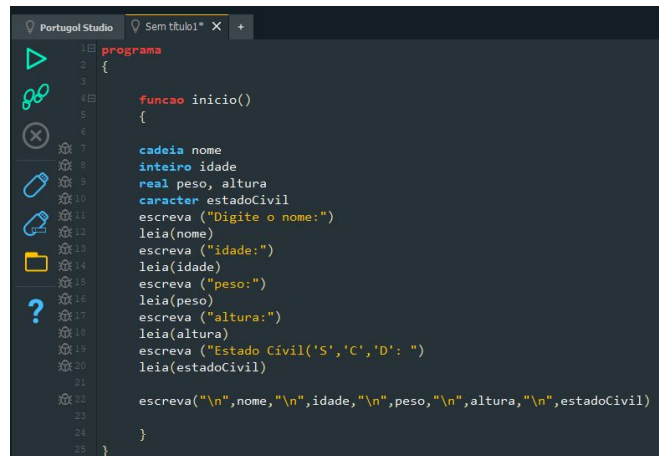
idade = 10

## Memória

idade	10
peso	
altura	

## Exercício:

Faça um programa para ler o nome, idade, peso, altura e estado cívil sendo do tipo caracter de uma pessoa e exibir os dados na tela.



```
1  programa
2  {
3
4      funcao inicio()
5      {
6
7          cadeia nome
8          inteiro idade
9          real peso, altura
10         caracter estadoCivil
11         escreva ("Digite o nome:")
12         leia(nome)
13         escreva ("idade:")
14         leia(idade)
15         escreva ("peso:")
16         leia(peso)
17         escreva ("altura:")
18         leia(altura)
19         escreva ("Estado Civil('S','C','D'): ")
20         leia(estadoCivil)
21
22         escreva("\n",nome,"\n",idade,"\n",peso,"\n",altura,"\n",estadoCivil)
23
24     }
25 }
```

# Operadores Matemáticos

+ soma

- subtração

\* multiplicação

/ divisão

% resto da divisão

# Comentários

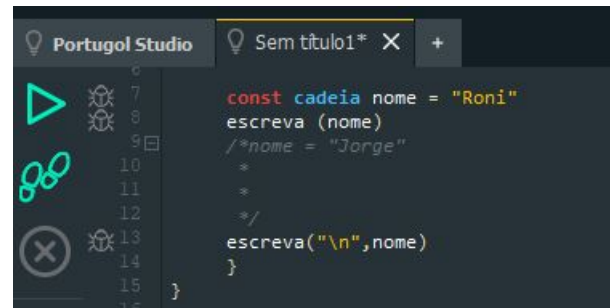
- Usado para ignorar uma parte do código ou para inserir informações sobre um determinado trecho de código



The screenshot shows the Portugol Studio IDE with a dark theme. The title bar includes 'Portugol Studio' and a tab titled 'Sem título1\*'. The code editor displays the following code:

```
6  
7 const cadeia nome = "Roni"  
8 escreva (nome)  
9 //nome = "Jorge"  
10 escreva("\n",nome)  
11 }  
12  
13  
14
```

On the left side of the editor, there are three icons: a green play button, a green infinity symbol, and a grey circle with an 'X'.



The screenshot shows the Portugol Studio IDE with a dark theme. The title bar includes 'Portugol Studio' and a tab titled 'Sem título1\*'. The code editor displays the following code:

```
6  
7 const cadeia nome = "Roni"  
8 escreva (nome)  
9 /*nome = "Jorge"  
10 *  
11 *  
12 */  
13 escreva("\n",nome)  
14 }  
15  
16
```

On the left side of the editor, there are three icons: a green play button, a green infinity symbol, and a grey circle with an 'X'.

# Teclas de Atalho

Pressione F11 para visualizar as teclas de atalho



	Ação	Atalho
📁	Abrir um arquivo	Ctrl+O
❓	Ajuda	F1
📄	Ativar/desativar a centralização de código fonte	Shift+Pause/Break
📄	Criar novo arquivo	Ctrl+N
🔍	Depurar programa atual	Shift+F5
🔍	Depurar próxima instrução do programa atual	Shift+F9
📄	Dicas de Interface	F3
▶	Executar programa atual	Shift+F6
📄	Exibir tela inicial	Alt+Home
🔍	Expandir/restaurar o tamanho do editor	Shift+Esc
✕	Fechar a aba atual	Ctrl+Q
✕	Fechar todas as abas	Ctrl+Shift+Q
⏏	Interromper a execução/depuração do programa atual	Shift+F7
🔍	Pesquisar e/ou substituir um texto no arquivo atual	Ctrl+F
💾	Salvar o arquivo atual	Ctrl+S
💾	Salvar uma cópia do arquivo atual	Ctrl+Shift+S
⏪	Selecionar a aba anterior (à esquerda)	Alt+Esquerda
⏩	Selecionar a próxima aba (à direita)	Alt+Direita

**Preenchimento Automático - CTRL + Espaço**

# Exercícios

- 1) Leia dois valores pelo teclado e imprima a soma.
- 2) Construir um algoritmo que leia um número e exiba na tela o seu sucessor e antecessor.
- 3) Construa um algoritmo que leia o nome de um aluno, disciplinas, duas notas e exiba na tela a média/
- 4) Faça um programa com duas variáveis ano\_nascimento que receberá o ano que você nasceu e outra variável com o nome ano\_futuro que deverá ser atribuído o valor 2035. Criar uma variável com o nome resultado para calcular a diferença. No final escreva na tela qual será a sua idade em 2035.

## Exercícios

5) Uma empresa paga R\$10.00 por hora normal trabalhada e R\$ 15.00 por hora extra. Escreva um algoritmo que leia o total de horas normais e o total de horas extras trabalhadas por um empregado em um ano e calcule o salário anual deste trabalhador.

*Exemplo : Entrada : Digite o número de horas trabalhadas no ano : 1760 Digite o número de horas extras trabalhadas no ano : 400*

*Saída : Seu salário anual é de : R\$ 23600*

6) Escreva um programa que receba a temperatura em Celsius e retorne o valor em Fahrenheit

*Exemplo : Entrada: Digite o valor da temperatura em Celsius : 10*

*Saída: 10 graus Celsius é o mesmo que 50 graus Fahrenheit*

7) Criar um algoritmo em português estruturado que leia dois números inteiros e imprima a seguinte saída: Dividendo, Divisor, Quociente e Resto

8) Escreva um programa que diga se o número é par ou ímpar Dica : o operador % calcula a divisão de um número X por um número Y e retorna o resto da divisão

Exemplo :

Entrada: Digite um número : 2

Saída: O número 2 é par